TRAITE D[illegible]OOPERATION EN MATIER[illegible]E BREVETS

Expéditeur: le BUREAU INTERNATIONAL

PCT

NOTIFICATION D'ELECTION

(règle 61.2 du PCT)

Destinataire:

Assistant Commissioner for Patents
United States Patent and Trademark
Office
Box PCT
Washington, D.C.20231
ETATS-UNIS D'AMERIQUE

en sa qualité d'office élu

Date d'expédition (jour/mois/année) 19 juin 2000 (19.06.00)	
Demande internationale no PCT/FR99/03065	**Référence du dossier du déposant ou du mandataire** 76.0552
Date du dépôt international (jour/mois/année) 08 décembre 1999 (08.12.99)	**Date de priorité** (jour/mois/année) 08 décembre 1998 (08.12.98)
Déposant BURIANNE, Yannick	

1. L'office désigné est avisé de son élection qui a été faite:

[X] dans la demande d'examen préliminaire international présentée à l'administration chargée de l'examen préliminaire international le:

15 mai 2000 (15.05.00)

[ ] dans une déclaration visant une élection ultérieure déposée auprès du Bureau international le:

2. L'élection [X] a été faite

[ ] n'a pas été faite

avant l'expiration d'un délai de 19 mois à compter de la date de priorité ou, lorsque la règle 32 s'applique, dans le délai visé à la règle 32.2b).

Bureau international de l'OMPI
34, chemin des Colombettes
1211 Genève 20, Suisse

no de télécopieur: (41-22) 740.14.35

Fonctionnaire autorisé

R. Forax

no de téléphone: (41-22) 338.83.38

TRAITE DE COOPERATION EN MATIERE DE BREVETS

PCT

RAPPORT D'EXAMEN PRELIMINAIRE INTERNATIONAL

4T

(article 36 et règle 70 du PCT)

Référence du dossier du déposant ou du mandataire 76.0552	**POUR SUITE A DONNER** voir la notification de transmission du rapport d'examen préliminaire international (formulaire PCT/IPEA/416)	
Demande internationale n° PCT/FR99/03065	Date du dépot international *(jour/mois/année)* 08/12/1999	Date de priorité *(jour/mois/année)* 08/12/1998
Classification internationale des brevets (CIB) ou à la fois classification nationale et CIB G07F7/10		
Déposant SCHLUMBERGER SYSTEMES et al.		

1. Le présent rapport d'examen préliminaire international, établi par l'administaration chargée de l'examen préliminaire international, est transmis au déposant conformément à l'article 36.

2. Ce RAPPORT comprend 5 feuilles, y compris la présente feuille de couverture.

   ☐ Il est accompagné d'ANNEXES, c'est-à-dire de feuilles de la description, des revendications ou des dessins qui ont été modifiées et qui servent de base au présent rapport ou de feuilles contenant des rectifications faites auprès de l'administration chargée de l'examen préliminaire international (voir la règle 70.16 et l'instruction 607 des Instructions administratives du PCT).

   Ces annexes comprennent feuilles.

3. Le présent rapport contient des indications relatives aux points suivants:

I	☒	Base du rapport
II	☐	Priorité
III	☐	Absence de formulation d'opinion quant à la nouveauté, l'activité inventive et la possibilité d'application industrielle
IV	☐	Absence d'unité de l'invention
V	☒	Déclaration motivée selon l'article 35(2) quant à la nouveauté, l'activité inventive et la possibilité d'application industrielle; citations et explications à l'appui de cette déclaration
VI	☐	Certains documents cités
VII	☒	Irrégularités dans la demande internationale
VIII	☐	Observations relatives à la demande internationale

Date de présentation de la demande d'examen préliminaire internationale	Date d'achèvement du présent rapport
15/05/2000	30.06.00
Nom et adresse postale de l'administration chargée de l'examen préliminaire international: Office européen des brevets D-80298 Munich Tél. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Fonctionnaire autorisé Beauce, G N° de téléphone +49 89 2399 2519

Formulaire PCT/IPEA/409 (feuille de couverture) (janvier 1994)

RAPPORT D'EXAMEN PRELIMINAIRE INTERNATIONAL

Demande internationale n° PCT/FR99/03065

I. Base du rapport

1. Ce rapport a été rédigé sur la base des éléments ci-après (*les feuilles de remplacement qui ont été remises à l'office récepteur en réponse à une invitation faite conformément à l'article 14 sont considérées, dans le présent rapport, comme "initialement déposées" et ne sont pas jointes en annexe au rapport puisqu'elles ne contiennent pas de modifications.*) :

Description, pages:

1-8 version initiale

Revendications, N°:

1-11 version initiale

Dessins, feuilles:

1/4-4/4 version initiale

2. Les modifications ont entrainé l'annulation :

☐ de la description, pages :
☐ des revendications, n^{os} :
☐ des dessins, feuilles :

3. ☐ Le présent rapport a été formulé abstraction faite (de certaines) des modifications, qui ont été considérées comme allant au-delà de l'exposé de l'invention tel qu'il a été déposé, comme il est indiqué ci-après (règle 70.2(c)) :

4. Observations complémentaires, le cas échéant :

V. Déclaration motivée selon l'article 35(2) quant à la nouveauté, l'activité inventive et la possibilité d'application industrielle; citations et explications à l'appui de cette déclaration

1. Déclaration

Nouveauté	Oui :	Revendications	1-11
	Non :	Revendications	
Activité inventive	Oui :	Revendications	1-11
	Non :	Revendications	
Possibilité d'application industrielle	Oui :	Revendications	1-11
	Non :	Revendications	

2. Citations et explications

voir feuille séparée

VII. Irrégularités dans la demande internationale

Les irrégularités suivantes, concernant la forme ou le contenu de la demande internationale, ont été constatées :

voir feuille séparée

Concernant le point V

Déclaration motivée selon l'article 35(2) quant à la nouveauté, l'activité inventive et la possibilité d'application industrielle; citations et explications à l'appui de cette déclaration

1. Il est fait référence au document suivant:
   D1: EP-A-0 540 095 (PHILIPS COMPOSANTS ;KONINKL PHILIPS ELECTRONICS NV (NL)) 5 mai 1993 (1993-05-05)

2. Le document D1 est considéré comme l'état de la technique le plus proche et décrit un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire.

   L'objet de la revendication indépendante 1 diffère de celui divulgué dans la document D1 en ce que le programme applicatif comprend au moins une variable configurable et une liste d'au moins un élément référence. De plus la mémoire comporte d'une part au moins un moyen d'initialisation de variables, ledit moyen étant paramétré par plusieurs paramètres dont l'un est la liste d'éléments références, et d'autre part, une commande permettant d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables.

   L'objet de la revendication indépendante 1 est donc nouveau au vu de l'article 33(2) PCT.
   La revendication de procédé correspondante satisfait également aux conditions énoncés à l'article 33(2) PCT.

3. L'inconvénient engendré par l'utilisation d'un tel dispositif, est l'emplacement mémoire important nécessaire pour l'initialisation des variables. De plus le temps nécessaire à l'exécution du programme applicatif est accru du fait de devoir l'exécuter même si les valeurs d'initialisation n'ont pas changées (car la phase d'initialisation fait partie intégrante du programme applicatif.

   La solution adoptée par la présente invention et contenue dans les revendications 1 et 11 ne découle pas de façon évidente de l'enseignement transmis par les documents cités dans le rapport de recherche international en combinaison avec

les connaissances de l'homme du métier.

Par conséquent l'objet des revendications indépendantes 1 et 11 satisfait aux conditions de l'article 33(3) PCT.

4. La condition d'application industrielle est également satisfaite (Article 33(4) PCT).

5. L'objet des revendications dépendantes 2 à 10 semble également satisfaire aux conditions de l'article 33 PCT.

Concernant le point VII

Irrégularités dans la demande internationale

1. Contrairement à ce qu'exige la règle 5.1 a) ii) PCT, la description n'indique pas l'état de la technique antérieure pertinent exposé dans le document D1 et ne cite pas ce document.

2. Les caractéristiques figurant dans les revendications ne comportent pas de signes de référence mis entre parenthèses (règle 6.2 b) PCT).

Translation.
09/857732

3

RECEIVED
NOV 30 2000
Technology Center 2100

PATENT COOPERATION TREATY

PCT

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70)

RECEIVED
DEC - 3 2001
2800 MAIL ROOM

Applicant's or agent's file reference 76.0552	**FOR FURTHER ACTION**	See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)
International application No. PCT/FR99/03065	International filing date (*day/month/year*) 08 December 1999 (08.12.99)	Priority date (*day/month/year*) 08 December 1998 (08.12.98)
International Patent Classification (IPC) or national classification and IPC G07F 7/10		
Applicant SCHLUMBERGER SYSTEMES		

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.

2. This REPORT consists of a total of 5 sheets, including this cover sheet.

☐ This report is also accompanied by ANNEXES, i.e., sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).

These annexes consist of a total of _____ sheets.

3. This report contains indications relating to the following items:

- I ☒ Basis of the report
- II ☐ Priority
- III ☐ Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
- IV ☐ Lack of unity of invention
- V ☒ Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement
- VI ☐ Certain documents cited
- VII ☒ Certain defects in the international application
- VIII ☐ Certain observations on the international application

Date of submission of the demand 15 May 2000 (15.05.00)	Date of completion of this report 30 June 2000 (30.06.2000)
Name and mailing address of the IPEA/EP Facsimile No.	Authorized officer Telephone No.

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No.
PCT/FR99/03065

I. Basis of the report

1. This report has been drawn on the basis of (*Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to the report since they do not contain amendments.*):

☐ the international application as originally filed.

☒ the description, pages 1-8, as originally filed,
pages _____, filed with the demand,
pages _____, filed with the letter of _____,
pages _____, filed with the letter of _____.

☒ the claims, Nos. 1-11, as originally filed,
Nos. _____, as amended under Article 19,
Nos. _____, filed with the demand,
Nos. _____, filed with the letter of _____,
Nos. _____, filed with the letter of _____.

☒ the drawings, sheets/fig 1/4-4/4, as originally filed,
sheets/fig _____, filed with the demand,
sheets/fig _____, filed with the letter of _____,
sheets/fig _____, filed with the letter of _____.

2. The amendments have resulted in the cancellation of:

☐ the description, pages _____
☐ the claims, Nos. _____
☐ the drawings, sheets/fig _____

3. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed, as indicated in the Supplemental Box (Rule 70.2(c)).

4. Additional observations, if necessary:

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Claims	1-11	YES
	Claims		NO
Inventive step (IS)	Claims	1-11	YES
	Claims		NO
Industrial applicability (IA)	Claims	1-11	YES
	Claims		NO

2. Citations and explanations

1. Reference is made to the following document:

D1: EP-A-0 540 095 (PHILIPS COMPOSANTS; KONINKL PHILIPS ELECTRONICS NV (NL)), May 5 1993 (1993-05-05).

2. Document D1, which is considered the closest prior art, describes an integrated circuit device with a memory and an application programme resident in said memory.

The subject matter of independent Claim 1 differs from that disclosed in document D1 in that said application programme includes at least one configurable variable and a list with at least one reference element. Moreover, said memory comprises at least one variable initialising means parameterised with a plurality of parameters, one of which is the list of reference elements, as well as an instruction for sending data containing, in particular, values to be assigned to said configurable variables.

The subject matter of independent Claim 1 is

therefore novel under PCT Article 33(2).

The corresponding method claim also meets the requirements of PCT Article 33(2).

3. The disadvantage resulting from the use of such a device is the substantial memory area required to initialise the variables. Moreover, the time required to run the application programme is increased due to the fact that it must be run even if the initialisation values have not been changed (since the initialisation phase is an integral part of the application programme).

The solution proposed by the present invention and contained in Claims 1 and 11 cannot be derived in an obvious manner from the teaching of the documents cited in the international search report, in combination with the knowledge of a person skilled in the art.

Hence, the subject matter of independent Claims 1 and 11 meets the requirements of PCT Article 33(3).

4. The requirements of industrial applicability are also met (PCT Article 33(4)).

5. The subject matter of dependent Claims 2 to 10 also appears to meet the requirements of PCT Article 33.

VII. Certain defects in the international application

The following defects in the form or contents of the international application have been noted:

1. Contrary to the requirements of PCT Rule 5.1(a)(ii), the description does not outline the relevant prior art set forth in document D1 and does not cite this document.

2. The features appearing in the claims are not accompanied by reference signs between brackets (PCT Rule 6.2(b)).

PCT ORGANISATION MONDIALE DE LA PROPRIETE INTELLECTUELLE
Bureau international

DEMANDE INTERNATIONALE PUBLIEE EN VERTU DU TRAITE DE COOPERATION EN MATIERE DE BREVETS (PCT)

(51) Classification internationale des brevets 7 : G07F 7/10	A1	(11) Numéro de publication internationale: WO 00/34927
		(43) Date de publication internationale: 15 juin 2000 (15.06.00)

(21) Numéro de la demande internationale: PCT/FR99/03065

(22) Date de dépôt international: 8 décembre 1999 (08.12.99)

(30) Données relatives à la priorité:
98/15493 8 décembre 1998 (08.12.98) FR

(71) Déposant *(pour tous les Etats désignés sauf US)*: SCHLUMBERGER SYSTEMES [FR/FR]; 50 avenue Jean Jaurès, F–92120 Montrouge (FR).

(72) Inventeur; et
(75) Inventeur/Déposant *(US seulement)*: BURIANNE, Yannick [FR/FR]; 1, allée des Noisetiers, F–92140 Clamart (FR).

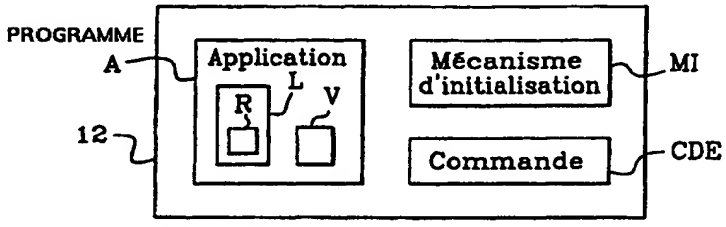
(74) Mandataire: UTZMANN–NORTH, Anne; Schlumberger Systemes, Test & Transactions, 50, avenue Jean Jaurès, Boîte postale 620–12, F–92542 Montrouge Cedex (FR).

(81) Etats désignés: CN, US, brevet européen (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Publiée
Avec rapport de recherche internationale.

(54) Title: DEVICE AND METHOD FOR INITIALISING AN APPLICATIVE PROGRAMME OF AN INTEGRATED CIRCUIT CARD

(54) Titre: DISPOSITIF ET PROCEDE D'INITIALISATION D'UN PROGRAMME APPLICATIF D'UNE CARTE A CIRCUIT INTEGRE



(57) Abstract

The invention concerns a device with an integrated circuit comprising a storage unit and a resident applicative programme in said storage unit. The invention is characterised in that said applicative programme comprises at least a configurable variable and a list of at least one reference element, and said storage unit comprises at least means for initialising said variables, said means being parameterised by several parameters whereof one of the parameters is said list of reference elements and a command for sending data containing in particular values to be assigned to the configurable variables. The invention is particularly applicable to chip cards.

(57) Abrégé

L'invention concerne un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire. L'invention se caractérise en ce que ledit programme applicatif comprend au moins une variable configurable et une liste d'au moins un élément référence, et en ce que ladite mémoire comporte, d'une part, au moins un moyen d'initialisation desdites variables, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références, et, d'autre part, une commande permettant d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables. L'invention s'applique, en particulier, aux cartes à puce.

UNIQUEMENT A TITRE D'INFORMATION

Codes utilisés pour identifier les Etats parties au PCT, sur les pages de couverture des brochures publiant des demandes internationales en vertu du PCT.

AL	Albanie	**ES**	Espagne	**LS**	Lesotho	**SI**	Slovénie
AM	Arménie	**FI**	Finlande	**LT**	Lituanie	**SK**	Slovaquie
AT	Autriche	**FR**	France	**LU**	Luxembourg	**SN**	Sénégal
AU	Australie	**GA**	Gabon	**LV**	Lettonie	**SZ**	Swaziland
AZ	Azerbaïdjan	**GB**	Royaume-Uni	**MC**	Monaco	**TD**	Tchad
BA	Bosnie-Herzégovine	**GE**	Géorgie	**MD**	République de Moldova	**TG**	Togo
BB	Barbade	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tadjikistan
BE	Belgique	**GN**	Guinée	**MK**	Ex-République yougoslave de Macédoine	**TM**	Turkménistan
BF	Burkina Faso	**GR**	Grèce			**TR**	Turquie
BG	Bulgarie	**HU**	Hongrie	**ML**	Mali	**TT**	Trinité-et-Tobago
BJ	Bénin	**IE**	Irlande	**MN**	Mongolie	**UA**	Ukraine
BR	Brésil	**IL**	Israël	**MR**	Mauritanie	**UG**	Ouganda
BY	Bélarus	**IS**	Islande	**MW**	Malawi	**US**	Etats-Unis d'Amérique
CA	Canada	**IT**	Italie	**MX**	Mexique	**UZ**	Ouzbékistan
CF	République centrafricaine	**JP**	Japon	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Pays-Bas	**YU**	Yougoslavie
CH	Suisse	**KG**	Kirghizistan	**NO**	Norvège	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	République populaire démocratique de Corée	**NZ**	Nouvelle-Zélande		
CM	Cameroun			**PL**	Pologne		
CN	Chine	**KR**	République de Corée	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Roumanie		
CZ	République tchèque	**LC**	Sainte-Lucie	**RU**	Fédération de Russie		
DE	Allemagne	**LI**	Liechtenstein	**SD**	Soudan		
DK	Danemark	**LK**	Sri Lanka	**SE**	Suède		
EE	Estonie	**LR**	Libéria	**SG**	Singapour		

DISPOSITIF ET PROCEDE D'INTIALISATION D'UN PROGRAMME APPLICATIF D'UNE CARTE A CIRCUIT INTEGRE

La présente invention concerne un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire. Elle concerne également un procédé d'initialisation d'un programme applicatif d'un tel dispositif.

Lesdits dispositifs sont en particulier des objets portatifs appelés cartes à puce comprenant des programmes applicatifs concernant le domaine de la santé, de la téléphonie mobile, ou encore, concernant le domaine bancaire.

Lesdites cartes à puce comportent un corps de carte dans lequel est intégré un module électronique contenant de manière classique un élément de commande (par exemple une unité centrale de traitement ou CPU) et une mémoire. Ladite mémoire comporte au moins un programme applicatif contenant des éléments unitaires auxquels on affecte des valeurs afin que le programme puisse être exécuté, lesdits éléments n'étant pas modifiés lors de l'exécution dudit programme applicatif. Ces éléments sont appelés variables configurables.

En vue de configurer lesdites variables, l'état de la technique propose des dispositifs qui prévoient des fichiers contenant des données qui sont affectées aux variables lors d'une phase dite d'initialisation. Cette phase d'initialisation est nécessaire au bon déroulement du programme applicatif. A cet effet, lesdits dispositifs comportent un moyen de commande qui permet de modifier les valeurs desdites données d'initialisation dans lesdits fichiers et ensuite d'affecter ces données auxdites variables. Lorsque ces variables sont stockées en mémoire de façon permanente, elles conservent leur valeur d'initialisation même si la carte n'est plus alimentée en tension.

Bien que ces dispositifs permettent de configurer un programme applicatif, les valeurs d'initialisation sont dupliquées dans deux espaces

mémoire de tailles quasi identiques, l'un contenant les fichiers de données d'initialisation et l'autre étant l'espace alloué pour les variables qui sont initialisées avec lesdites données, ce qui peut être gênant du fait de la taille limitée de la mémoire des cartes à puce. De plus, le temps d'exécution dudit programme applicatif est sensiblement accru du fait notamment de la nécessité d'effectuer ladite phase d'initialisation lors de chaque exécution du programme même si les valeurs d'initialisation n'ont pas changées car ladite phase d'initialisation fait partie intégrante du programme applicatif. Enfin, il existe des cas où, soit le programme applicatif ne possède aucun privilège pour accéder auxdits fichiers, soit ladite carte ne possède tout simplement aucun fichier.

Aussi un problème technique à résoudre par l'objet de la présente invention est de proposer un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire, ainsi qu'un procédé d'initialisation d'un programme applicatif d'un tel dispositif, qui permettraient, d'une part, de configurer un programme applicatif sans avoir de duplication de données et ainsi éviter des pertes d'espace mémoire dues aux fichiers précités, et, d'autre part, d'éviter d'augmenter le temps d'exécution dudit programme applicatif .

Une solution au problème technique posé se caractérise, selon un premier objet de la présente invention, en ce que ledit programme applicatif comprend au moins une variable configurable et une liste d'au moins un élément référence, et en ce que ladite mémoire comporte, d'une part, au moins un moyen d'initialisation desdites variables, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références, et, d'autre part, une commande permettant d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables.

Selon un second objet de la présente invention, cette solution se caractérise en ce que le procédé d'initialisation comporte les étapes consistant à :

-créer, dans ledit programme applicatif, au moins une variable configurable et une liste d'au moins un élément référence,

-envoyer des données contenant en particulier des valeurs à affecter aux variables configurables,

-initialiser lesdites variables grâce à un moyen d'initialisation, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références.

Ainsi, comme on le verra en détail plus loin, le dispositif de l'invention permet d'avoir une gestion optimisée de la mémoire de la carte et une configuration directe des variables d'un programme applicatif grâce à la commande qui permet de modifier les valeurs affectées aux variables configurables et grâce également à la liste d'éléments références passée en paramètre du moyen d'initialisation, liste qui permet d'établir un lien entre les valeurs envoyées par ladite commande et les variables du programme applicatif à configurer.

La description qui va suivre au regard des dessins annexés, donnée à titre d'exemple non limitatif, fera bien comprendre en quoi consiste l'invention et comment elle peut être réalisée.

La figure 1 est un schéma d'un dispositif à circuit intégré, ici une carte à puce.

La figure 2 est un schéma d'une mémoire de la carte de la figure 1.

La figure 3 est un schéma d'un programme applicatif de la carte de la figure 1.

La figure 4 est un schéma d'une commande de la carte de la figure 1.

La figure 5 est un schéma d'une liste d'éléments d'un programme applicatif de la mémoire de la figure 2.

La figure 6 est un autre schéma de la mémoire de la carte de la figure 1.

La figure 7 est un schéma montrant des variables contenues dans le programme applicatif de la figure 3.

Sur la figure 1 est représenté un dispositif 10 à circuit intégré, ici une carte à puce.

Cette carte 10 contient un élément 11 de commande (par exemple une unité centrale de traitement ou CPU), une mémoire 12 et un bloc 13 de contacts destiné à une connexion électrique avec par exemple un connecteur d'un lecteur de cartes.

Ladite mémoire 12 est représentée sur la figure 2. Elle comprend un programme applicatif A. Ledit programme A comprend au moins une variable configurable V et une liste L d'au moins un élément référence R. Ladite mémoire comporte, d'une part, au moins un moyen MI d'initialisation desdites variables V, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste L d'éléments références, et, d'autre part, une commande CDE permettant d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables. Le moyen MI est une fonction ou un bout de programme. Sur la figure 3, le programme applicatif A comporte trois variables configurables V1, V2 et V3 et une liste L contenant trois éléments références R1, R2 et R3.

Afin que le programme A se déroule correctement, il faut configurer ses variables, c'est à dire leur affecter des valeurs.

Dans une première étape, la commande CDE est envoyée à la carte 10. Elle comporte des données telles que par exemple, un nombre d'éléments références R, des numéros indexant les éléments références d'une liste, des valeurs associées.... Sur la figure 4, la commande CDE

envoie les trois valeurs alphanumériques suivantes APPLICATION GSM, TELEPHONER et APPEL EN COURS. Ces valeurs sont précédées des index 1, 2 et 3 qui correspondent à trois éléments références.

Lorsque le programme applicatif A reçoit la commande CDE, il est exécuté et la phase d'initialisation faisant appel au moyen MI commence.

Dans une deuxième étape, on construit un lien entre les valeurs envoyées par la commande CDE et les éléments références d'une liste L spécifique. La liste L d'éléments références paramétrant le moyen MI d'initialisation permet d'établir ce lien. Les autres paramètres sont entre autres les données envoyées par ladite commande CDE. On spécifie la liste L en donnant par exemple son nom. Sur la figure 5, L est appelée CUSTOMELEMENT. Elle contient trois éléments références MENU, TEXT et MESSAGE auxquels sont associées les valeurs alphanumériques respectives APPLICATION GSM, TELEPHONER et APPEL EN COURS. Ces valeurs proviennent de la commande CDE.

Dans une troisième étape, le moyen MI d'initialisation établit un lien entre lesdites valeurs de ladite liste L et les variables à configurer V grâce aux éléments références R. A cet effet, un élément référence R fait référence à une variable configurable V. Sur la figure 3, R1, R2 et R3 font respectivement référence aux variables V1, V2 et V3, ces dernières étant des variables dont on veut initialiser tout ou partie de leur contenu. C'est grâce à ces différents liens que le transfert des valeurs vers lesdites variables s'effectue.

Une fois ce transfert effectué, la configuration du programme applicatif A est terminée et la suite dudit programme peut se dérouler comme souhaité. Le dispositif selon l'invention ne comporte aucun fichier, de ce fait, la configuration des variables a été directe.

On notera que l'invention prévoit également que ladite commande CDE permet de lire le contenu des variables configurables et ce grâce à

la présence d'un paramètre de ladite commande appelé MODE qui indique si la commande doit envoyer ou lire des données. Ceci permet de lire les valeurs des variables V à tout moment et par conséquent de connaître la configuration du programme applicatif A à tout moment.

Il peut être intéressant pour gagner de l'espace mémoire et homogénéiser les initialisations de permettre à un ou plusieurs programmes applicatifs d'utiliser le même moyen MI d'initialisation. Aussi, au moins un moyen MI d'initialisation réside dans ladite mémoire indépendamment d'un programme applicatif A. Cela signifie que ledit moyen MI peut être utilisé par tous les programmes applicatifs résidents dans la carte 10 et n'est propre à aucun programme A en particulier. Comme le montre la figure 6, le moyen MI1 est indépendant des programmes applicatifs A1 et A2 et peut être de ce fait utilisé par l'un ou l'autre de ces programmes.

Cependant, il peut être également utile de pouvoir personnaliser le moyen d'initialisation pour un programme applicatif donné en ayant un moyen différent de MI1, par exemple dans le cas où l'on veut avoir un protocole d'échange de données différent de celui de MI1 c'est à dire un format de données d'initialisation différent. Comme le montre la figure 6, au moins un programme applicatif A2 comprend un moyen MI2 d'initialisation. Pour configurer les variables de A2, on aura le choix d'utiliser les moyens MI1 ou MI2 si lesdites variables respectent le format de données respectif desdits moyens.

On notera que l'on peut également n'avoir aucun moyen MI indépendant d'un programme applicatif, chaque moyen MI d'initialisation étant, dans ce cas, propre à un programme applicatif, ou au contraire n'avoir que des moyens indépendants.

La présente invention s'applique particulièrement à des programmes applicatifs qui sont programmés dans des langages de haut niveau tels qu'en particulier un langage appelé JAVA (marque

déposée). Ce langage traite des notions de classe, d'héritage, d'attribut et de méthode bien connues de l'homme du métier.

Dans le cas où le programme applicatif A est programmé en JAVA, les variables configurables sont des objets et une liste d'éléments références fait référence à un ensemble d'objets. Sur la figure 7, la mémoire 12 comprend un programme applicatif A. Ledit programme applicatif A comporte au moins deux variables V1 et V3 configurables référencées dans une même liste et qui dérivent d'une même classe mère C0. De plus, ledit programme applicatif A comporte au moins deux variables V1 et V2 configurables référencées dans une même liste et qui sont des instances d'une même classe C1. Les différentes classes sont définies soit dans le programme applicatif A, soit de façon indépendante, par exemple dans une librairie. Lesdites variables configurables sont persistantes dans ladite mémoire 12.

On peut voir que ladite liste L représente des objets ayant, soit des points en commun, les variables ou objets V1 et V3 héritent de l'attribut At1 et des méthodes M1 et M2 de la classe C0 mais ont leur propres attributs et méthodes, soit tous leurs points en commun, V1 et V2 sont des instances de la classe C1 qui possède l'attribut At2 et la méthode M3. Pour configurer lesdits objets, il faut qu'une liste L soit du même type qu'une classe mère ou que la classe desdits objets. Ainsi un moyen MI1 simple permettra de configurer une partie du contenu des objets V1, V2 et V3, soit l'attribut At1. On pourrait également avoir un autre moyen MI2 plus complexe permettant de configurer l'ensemble des attributs At1 et At2 des variables V1 et V2.

C'est grâce à la définition du type de ladite liste L que la présente invention nous permet de modifier les valeurs des attributs d'objets bien spécifiés et d'empêcher ainsi la modification par inadvertance du contenu d'autres objets. De plus, grâce à la présente invention, il n'y a pas d'accès direct à l'emplacement mémoire contenant toutes les

variables du programme applicatif A et, par suite, on ne risque pas de modifier de façon frauduleuse toutes ces variables.

Un autre avantage de la présente invention est que lesdites variables ou objets sont persistants en mémoire. Cela signifie qu'une fois configurés et lorsqu'ils ne sont pas modifiés pendant l'exécution du programme applicatif A, lesdits objets conservent leurs valeurs d'initialisation même après l'exécution du programme A. Si on ne veut pas modifier ces valeurs avant une autre exécution de A, il est inutile pour un utilisateur d'envoyer la commande CDE pour reconfigurer le programme applicatif A. Par suite, on s'affranchit de la phase d'initialisation et aucun moyen MI d'initialisation n'est déclenché. Par conséquent, le temps d'exécution est diminué.

Comme nous venons de le voir, le langage JAVA est intéressant à plus d'un égard, mais une de ses caractéristiques qui fait également sa force est qu'il possède des moyens sécuritaires dont un moyen qui vérifie que chaque instruction d'un programme applicatif A est valide ainsi que les paramètres de cette instruction. Par exemple, si une instruction nécessite un tableau d'octets situé à une certaine adresse de la mémoire 12 comme paramètre alors qu'une adresse mémoire interdite est désignée à la place, ledit moyen sécuritaire permettra de détecter cette erreur et d'empêcher ainsi l'accès à un espace mémoire interdit. Afin de profiter de ces moyens sécuritaires de vérification, l'invention prévoit que tout moyen MI d'initialisation est défini dans le même langage que ledit programme applicatif A, c'est à dire en JAVA. Ainsi, si un paramètre dudit moyen MI est faux, le programme ne sera pas exécuté et un fraudeur ne pourra accéder à des emplacements mémoire interdits.

REVENDICATIONS

1 - Dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire, caractérisé en ce que ledit programme applicatif comprend au moins une variable configurable et une liste d'au moins un élément référence, et en ce que ladite mémoire comporte, d'une part, au moins un moyen d'initialisation desdites variables, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références, et, d'autre part, une commande permettant d'envoyer des données contenant en particulier des valeurs à affecter aux variables configurables.

2 - Dispositif selon la revendication 1, caractérisé en ce que lesdites variables configurables sont persistantes dans ladite mémoire.

3 - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'un élément référence fait référence à une variable configurable.

4 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ledit programme applicatif comporte au moins deux variables configurables référencées dans une même liste et qui dérivent d'une même classe mère.

5 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ledit programme applicatif comporte au moins deux variables configurables référencées dans une même liste et qui sont des instances d'une même classe.

6 - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'au moins un moyen d'initialisation réside dans ladite mémoire indépendamment d'un programme applicatif.

7 - Dispositif selon l'une des revendications précédentes, caractérisé en ce qu'au moins un programme applicatif comprend un moyen d'initialisation.

8 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que tout moyen d'initialisation est défini dans le même langage que ledit programme applicatif.

9 - Dispositif selon l'une des revendications précédentes, caractérisé en ce que ladite commande permet de lire le contenu des variables configurables.

10 – Procédé d'initialisation d'un programme applicatif d'un dispositif à circuit intégré comprenant une mémoire et au moins un programme applicatif résident dans ladite mémoire, caractérisé en ce que ledit procédé comporte les étapes consistant à :

-créer, dans ledit programme applicatif, au moins une variable configurable et une liste d'au moins un élément référence,

-envoyer des données contenant en particulier des valeurs à affecter aux variables configurables,

-initialiser lesdites variables grâce à un moyen d'initialisation, ledit moyen étant paramétré par plusieurs paramètres dont l'un des paramètres est ladite liste d'éléments références.

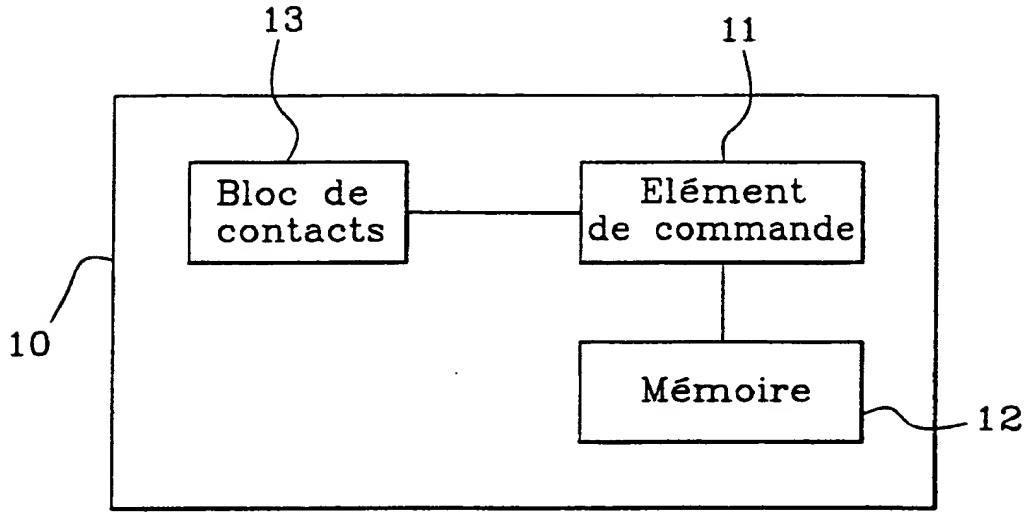


FIG.1

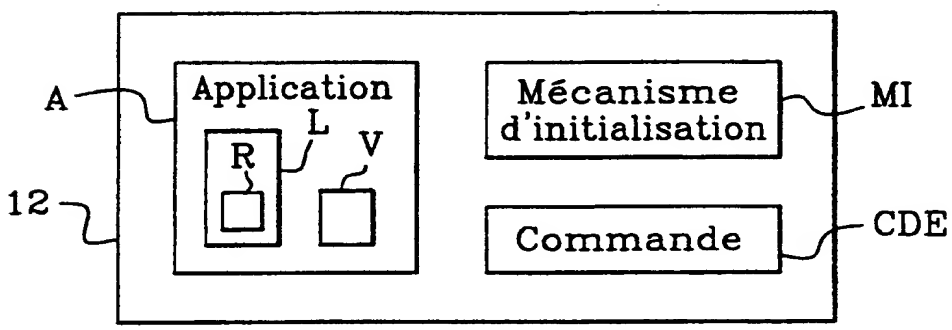


FIG.2

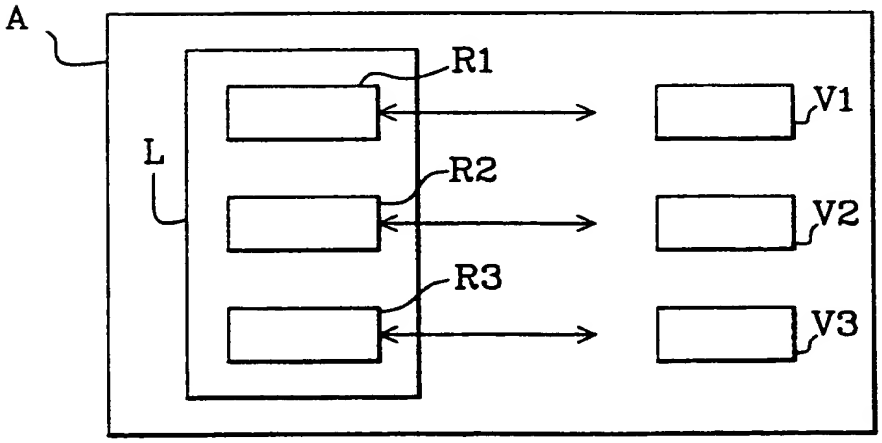


FIG.3

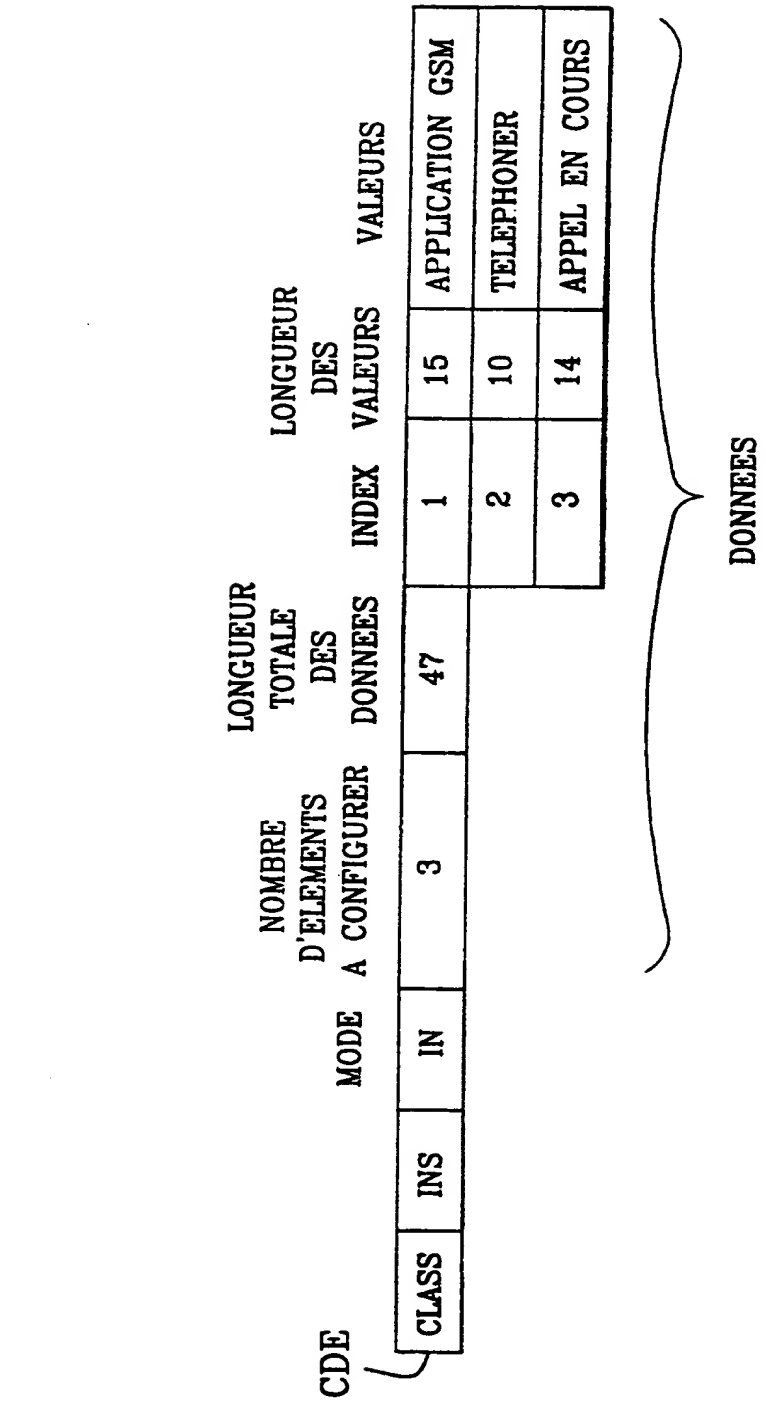

CDE
CLASS
INS
MODE
IN
NOMBRE D'ELEMENTS A CONFIGURER
3
LONGUEUR TOTALE DES DONNEES
47
INDEX
LONGUEUR DES VALEURS
VALEURS
1
15
APPLICATION GSM
2
10
TELEPHONER
3
14
APPEL EN COURS
DONNEES

FIG.4

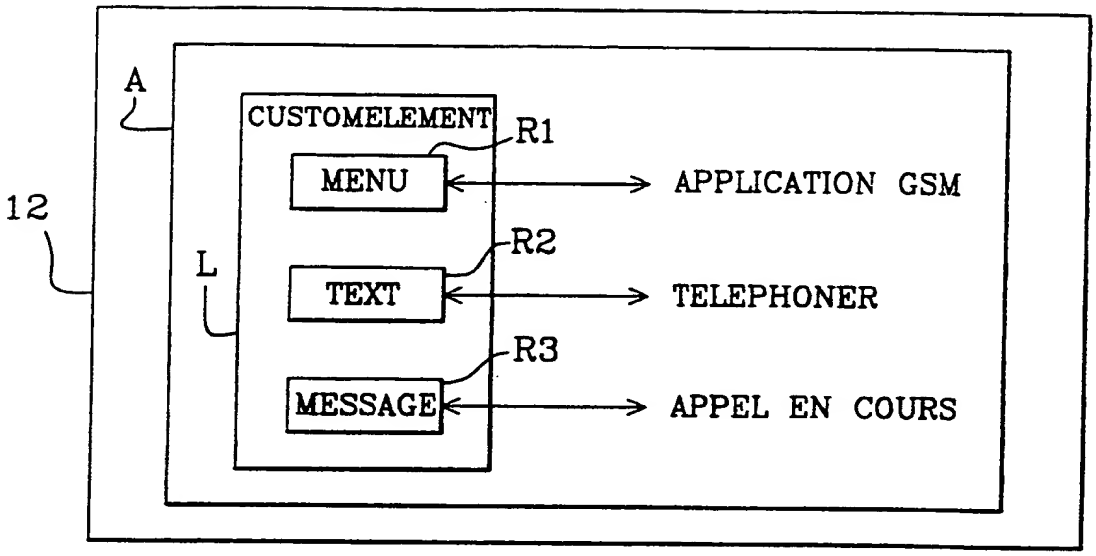

A
CUSTOMELEMENT
R1
MENU
APPLICATION GSM
12
L
R2
TEXT
TELEPHONER
R3
MESSAGE
APPEL EN COURS

FIG.5

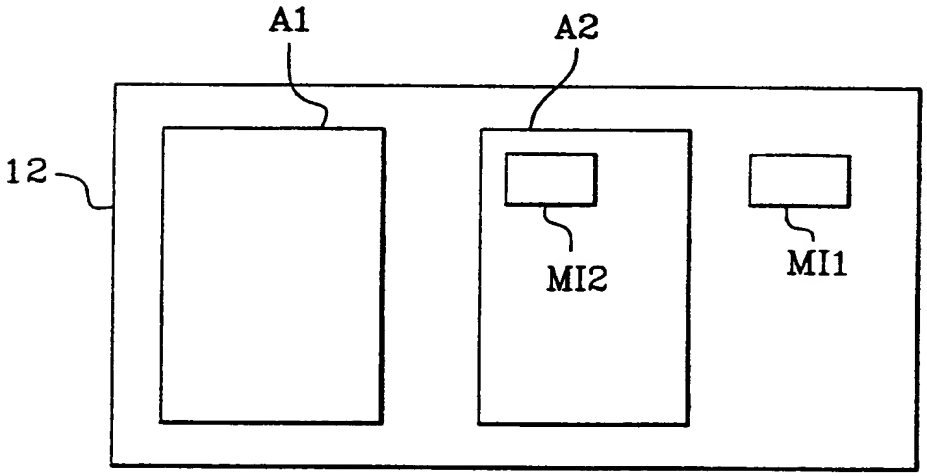

A1
A2
12
MI2
MI1

FIG.6

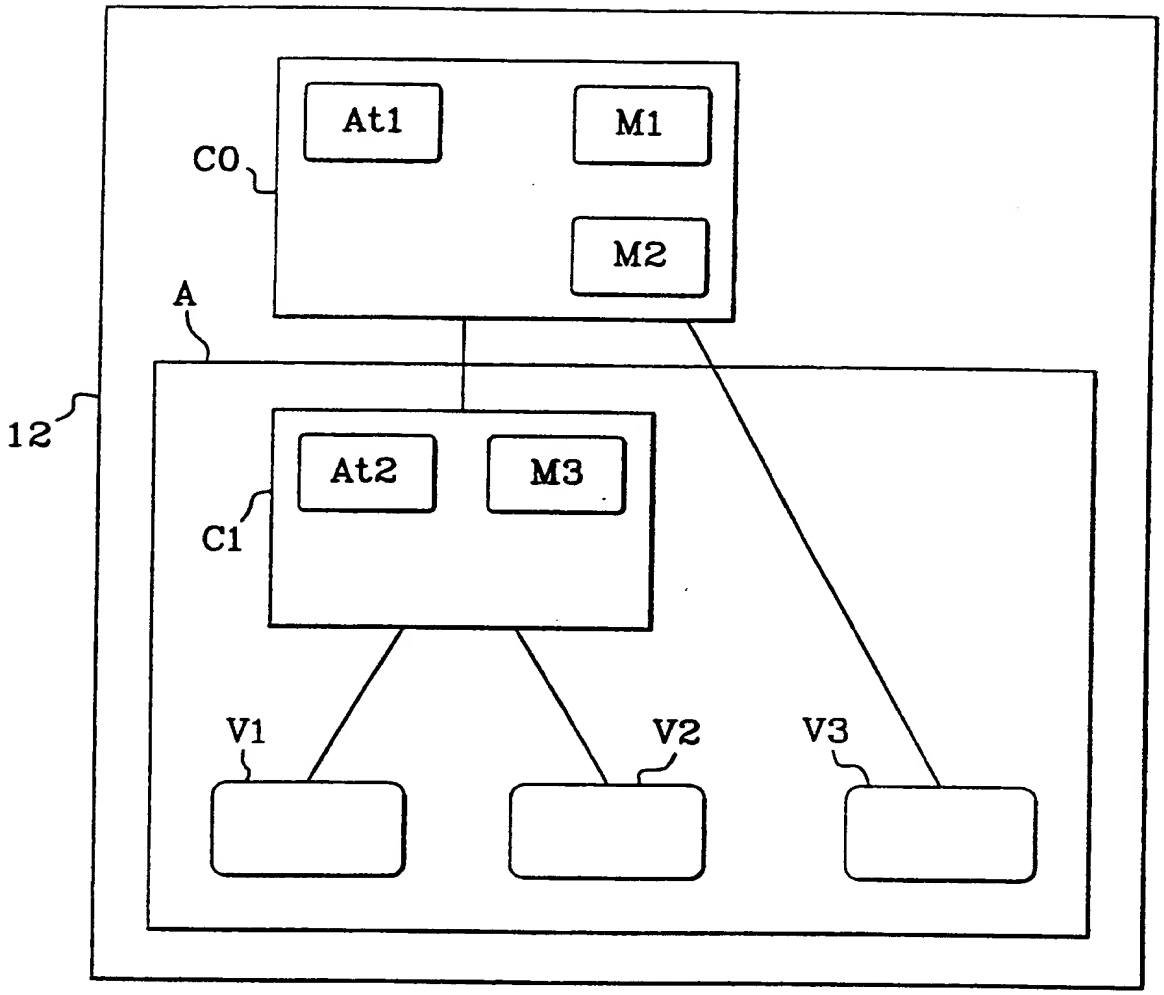


FIG.7

TRAITE [redacted]OOPERATION EN MATIERE DE[redacted]EVETS

PCT

RAPPORT DE RECHERCHE INTERNATIONALE

(article 18 et règles 43 et 44 du PCT)

Référence du dossier du déposant ou du mandataire 76.0552	**POUR SUITE A DONNER** voir la notification de transmission du rapport de recherche internationale (formulaire PCT/ISA/220) et, le cas échéant, le point 5 ci-après	
Demande internationale n° PCT/FR 99/ 03065	Date du dépôt international *(jour/mois/année)* 08/12/1999	(Date de priorité (la plus ancienne) *(jour/mois/année)* 08/12/1998
Déposant SCHLUMBERGER SYSTEMES et al.		

Le présent rapport de recherche internationale, établi par l'administration chargée de la recherche internationale, est transmis au déposant conformément à l'article 18. Une copie en est transmise au Bureau international.

Ce rapport de recherche internationale comprend 3 feuilles.

[X] Il est aussi accompagné d'une copie de chaque document relatif à l'état de la technique qui y est cité.

1. **Base du rapport**

   a. En ce qui concerne la **langue**, la recherche internationale a été effectuée sur la base de la demande internationale dans la langue dans laquelle elle a été déposée, sauf indication contraire donnée sous le même point.

   [ ] la recherche internationale a été effectuée sur la base d'une traduction de la demande internationale remise à l'administration.

   b. En ce qui concerne **les séquences de nucléotides ou d'acides aminés** divulguées dans la demande internationale (le cas échéant), la recherche internationale a été effectuée sur la base du listage des séquences :

   [ ] contenu dans la demande internationale, sous forme écrite.

   [ ] déposée avec la demande internationale, sous forme déchiffrable par ordinateur.

   [ ] remis ultérieurement à l'administration, sous forme écrite.

   [ ] remis ultérieurement à l'administration, sous forme déchiffrable par ordinateur.

   [ ] La déclaration, selon laquelle le listage des séquences présenté par écrit et fourni ultérieurement ne vas pas au-delà de la divulgation faite dans la demande telle que déposée, a été fournie.

   [ ] La déclaration, selon laquelle les informations enregistrées sous forme déchiffrable par ordinateur sont identiques à celles du listage des séquences présenté par écrit, a été fournie.

2. [ ] **Il a été estimé que certaines revendications ne pouvaient pas faire l'objet d'une recherche** (voir le cadre I).

3. [ ] **Il y a absence d'unité de l'invention** (voir le cadre II).

4. En ce qui concerne le **titre**,

   [X] le texte est approuvé tel qu'il a été remis par le déposant.

   [ ] Le texte a été établi par l'administration et a la teneur suivante:

5. En ce qui concerne **l'abrégé**,

   [X] le texte est approuvé tel qu'il a été remis par le déposant

   [ ] le texte (reproduit dans le cadre III) a été établi par l'administration conformément à la règle 38.2b). Le déposant peut présenter des observations à l'administration dans un délai d'un mois à compter de la date d'expédition du présent rapport de recherche internationale.

6. La figure **des dessins** à publier avec l'abrégé est la Figure n° 2

   [X] suggérée par le déposant.

   [ ] parce que le déposant n'a pas suggéré de figure.

   [ ] parce que cette figure caractérise mieux l'invention.

   [ ] Aucune des figures n'est à publier.

INTERNATIONAL SEARCH REPORT

International Application No
PCT/FR 99/03065

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G07F7/10

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G07F G06K

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 540 095 A (PHILIPS COMPOSANTS ;KONINKL PHILIPS ELECTRONICS NV (NL)) 5 May 1993 (1993-05-05) abstract column 3, line 10 -column 4, line 19 column 7, line 27 -column 8, line 20 column 12, line 23 -column 14, line 31 claim 1; figure 1	1-10
A	FR 2 759 795 A (FRANCOIS CHARLES OBERTHUR FIDU) 21 August 1998 (1998-08-21) page 2, line 10 -page 3, line 8 page 4, line 6 - line 31 claim 7; figures 1,2	1-10

-/—

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
16 March 2000	27/03/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Miltgen, E

Form PCT/ISA/210 (second sheet) (July 1992)

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 515 532 A (IIJIMA YASUO) 7 May 1996 (1996-05-07) abstract column 1, line 60 -column 4, line 47 claims 1-5; figures 2,15	1,10
A	US 5 019 970 A (YAMAGUCHI ATSUO ET AL) 28 May 1991 (1991-05-28) abstract column 2, line 39 -column 3, line 66 claims 1,2; figures 1,2	1,10
A	PATENT ABSTRACTS OF JAPAN vol. 098, no. 003, 27 February 1998 (1998-02-27) & JP 09 305734 A (DAINIPPON PRINTING CO LTD), 28 November 1997 (1997-11-28) abstract	1,10
A	EP 0 674 290 A (FUJITSU LTD) 27 September 1995 (1995-09-27)	

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/FR 99/03065

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0540095 A	05-05-1993	FR 2683357 A	07-05-1993
		DE 69223920 D	12-02-1998
		DE 69223920 T	18-06-1998
		JP 5217035 A	27-08-1993
		US 5452431 A	19-09-1995
FR 2759795 A	21-08-1998	WO 9836387 A	20-08-1998
US 5515532 A	07-05-1996	JP 7093203 A	07-04-1995
US 5019970 A	28-05-1991	JP 2012486 A	17-01-1990
		JP 2501874 B	29-05-1996
		DE 3844032 A	04-01-1990
		FR 2633756 A	05-01-1990
JP 09305734 A	28-11-1997	NONE	
EP 0674290 A	27-09-1995	JP 7239928 A	12-09-1995
		US 5506397 A	09-04-1996
		US 5563395 A	08-10-1996

RAPPORT DE RECHERCHE INTERNATIONALE

Demande internationale No
PCT/FR 99/03065

A. CLASSEMENT DE L'OBJET DE LA DEMANDE
CIB 7 G07F7/10

Selon la classification internationale des brevets (CIB) ou à la fois selon la classification nationale et la CIB

B. DOMAINES SUR LESQUELS LA RECHERCHE A PORTE

Documentation minimale consultée (système de classification suivi des symboles de classement)
CIB 7 G07F G06K

Documentation consultée autre que la documentation minimale dans la mesure où ces documents relèvent des domaines sur lesquels a porté la recherche

Base de donnée électronique consultée au cours de la recherche internationale (nom de la base de données, et si réalisable, termes de recherche utilisés)

C. DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie °	Identification des documents citée, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées
A	EP 0 540 095 A (PHILIPS COMPOSANTS ;KONINKL PHILIPS ELECTRONICS NV (NL)) 5 mai 1993 (1993-05-05) abrégé colonne 3, ligne 10 -colonne 4, ligne 19 colonne 7, ligne 27 -colonne 8, ligne 20 colonne 12, ligne 23 -colonne 14, ligne 31 revendication 1; figure 1	1-10
A	FR 2 759 795 A (FRANCOIS CHARLES OBERTHUR FIDU) 21 août 1998 (1998-08-21) page 2, ligne 10 -page 3, ligne 8 page 4, ligne 6 - ligne 31 revendication 7; figures 1,2	1-10

-/—

[X] Voir la suite du cadre C pour la fin de la liste des documents

[X] Les documents de familles de brevets sont indiqués en annexe

° Catégories spéciales de documents citées:

"A" document définissant l'état général de la technique, non considéré comme particulièrement pertinent

"E" document antérieur, mais publié à la date de dépôt international ou après cette date

"L" document pouvant jeter un doute sur une revendication de priorité ou cité pour déterminer la date de publication d'une autre citation ou pour une raison spéciale (telle qu'indiquée)

"O" document se référant à une divulgation orale, à un usage, à une exposition ou tous autres moyens

"P" document publié avant la date de dépôt international, mais postérieurement à la date de priorité revendiquée

"T" document ultérieur publié après la date de dépôt international ou la date de priorité et n'appartenenant pas à l'état de la technique pertinent, mais cité pour comprendre le principe ou la théorie constituant la base de l'invention

"X" document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme nouvelle ou comme impliquant une activité inventive par rapport au document considéré isolément

"Y" document particulièrement pertinent; l'invention revendiquée ne peut être considérée comme impliquant une activité inventive lorsque le document est associé à un ou plusieurs autres documents de même nature, cette combinaison étant évidente pour une personne du métier

"&" document qui fait partie de la même famille de brevets

Date à laquelle la recherche internationale a été effectivement achevée	Date d'expédition du présent rapport de recherche internationale
16 mars 2000	27/03/2000

Nom et adresse postale de l'administration chargée de la recherche internationale	Fonctionnaire autorisé
Office Européen des Brevets, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Miltgen, E

1

Formulaire PCT/ISA/210 (deuxième feuille) (Juillet 1992)

C.(suite) DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie *	Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées
A	US 5 515 532 A (IIJIMA YASUO) 7 mai 1996 (1996-05-07) abrégé colonne 1, ligne 60 -colonne 4, ligne 47 revendications 1-5; figures 2,15	1,10
A	US 5 019 970 A (YAMAGUCHI ATSUO ET AL) 28 mai 1991 (1991-05-28) abrégé colonne 2, ligne 39 -colonne 3, ligne 66 revendications 1,2; figures 1,2	1,10
A	PATENT ABSTRACTS OF JAPAN vol. 098, no. 003, 27 février 1998 (1998-02-27) & JP 09 305734 A (DAINIPPON PRINTING CO LTD), 28 novembre 1997 (1997-11-28) abrégé	1,10
A	EP 0 674 290 A (FUJITSU LTD) 27 septembre 1995 (1995-09-27)	

RAPPORT DE RECHERCHE INTERNATIONALE

Renseignements relatifs aux membres de familles de brevets

Demande internationale No
PCT/FR 99/03065

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
EP 0540095 A	05-05-1993	FR 2683357 A	07-05-1993
		DE 69223920 D	12-02-1998
		DE 69223920 T	18-06-1998
		JP 5217035 A	27-08-1993
		US 5452431 A	19-09-1995
FR 2759795 A	21-08-1998	WO 9836387 A	20-08-1998
US 5515532 A	07-05-1996	JP 7093203 A	07-04-1995
US 5019970 A	28-05-1991	JP 2012486 A	17-01-1990
		JP 2501874 B	29-05-1996
		DE 3844032 A	04-01-1990
		FR 2633756 A	05-01-1990
JP 09305734 A	28-11-1997	AUCUN	
EP 0674290 A	27-09-1995	JP 7239928 A	12-09-1995
		US 5506397 A	09-04-1996
		US 5563395 A	08-10-1996

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Numéro de publication: **0 540 095 A1**

(12) DEMANDE DE BREVET EUROPEEN

(21) Numéro de dépôt: **92203251.1**

(22) Date de dépôt: **22.10.92**

(51) Int. Cl.5: **G06K 19/073**, G07F 7/10, G06F 12/14

(30) Priorité: **30.10.91 FR 9113409**

(43) Date de publication de la demande: **05.05.93 Bulletin 93/18**

(84) Etats contractants désignés: **DE FR GB**

(71) Demandeur: **PHILIPS COMPOSANTS**
117, quai du Président Roosevelt
F-92130 Issy les Moulineaux(FR)
(84) **FR**

(71) Demandeur: **N.V. Philips' Gloeilampenfabrieken**
Groenewoudseweg 1
NL-5621 BA Eindhoven(NL)
(84) **DE GB**

(72) Inventeur: **Bournas, Jean-Pierre**
Société Civile S.P.I.D., 156, Boulevard Haussmann
F-75008 Paris(FR)

(74) Mandataire: **Pinchon, Pierre et al**
Société Civile S.P.I.D. 156, Boulevard Haussmann
F-75008 Paris (FR)

(54) Microcircuit pour carte à puce à mémoire programmable protégée.

(57) Microcircuit pour carte à puce multi-applications comportant notamment une mémoire ROM (12), une mémoire programmable (13) et un circuit de contrôle d'adressage (14) de cette mémoire programmable.

Selon l'invention, la mémoire programmable (13) est partagée en au moins une zone de répertoire (ZR) et une zone des applications (ZA), la zone de répertoire (ZR) comportant par application chargée, au moins un code de référence d'une application i présente dans la zone ZA_i ainsi que les adresses ZA_{il} et ZA_{ih} respectivement de début et de fin de la zone ZA_i allouée à cette application.

Le microcircuit comporte également des moyens (25 à 50) pour inhiber toute commande (R, W ou E) de la mémoire programmable (13) lorsqu'elle est relative à une adresse extérieure à l'intervalle ZA_{il} - ZA_{ih} de l'application en cours de traitement et sauf s'il s'agit d'une opération prioritaire spécifiquement prévue par un programme fixé dans la mémoire ROM (12).

Application : carte à microcircuits.

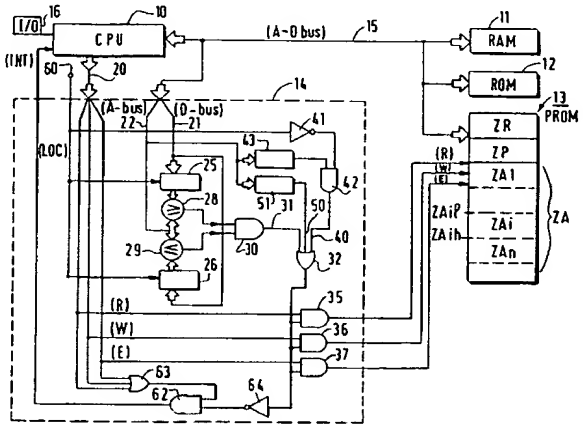


FIG.1

EP 0 540 095 A1

La présente invention concerne un microcircuit pour carte à puce comprenant entre-autres : des moyens d'accès entrée-sortie, au moins une mémoire qui est programmable et accessible par un espace d'adressage à adresses consécutives, un circuit de contrôle d'adressage de cette mémoire programmable déterminant soit une inhibition, soit une validation des commandes d'écriture et/ou de lecture par la comparaison de l'adresse demandée avec deux valeurs limites de début et de fin d'une zone particulière d'adresses, limites qui sont mémorisées au sein du circuit de contrôle d'adressage lequel circuit comporte également une voie d'autorisation prioritaire permettant de s'affranchir temporairement de ladite inhibition dans des circonstances prédéterminées et particulières.

Un tel microcircuit est notamment connu du document FR 2 304 989.

L'application dans le domaine bancaire des cartes portatives au sein desquelles est incorporé un circuit intégré électronique (encore appelé "puce") est bien connue ainsi que la grande sécurité de fonctionnement qui en résulte. Il est en effet pratiquement impossible d'accéder à certaines données inscrites dans des zones protégées du microcircuit sans le détruire. Par ailleurs, lors de l'utilisation, un protocole d'identification faisant intervenir un code personnel et secret, inscrit dans une zone protégée du microcircuit, permet d'interdire toute tentative d'usage de la carte si les conditions d'identification ne sont pas satisfaites.

Dans le cas d'une carte bancaire, le fabricant de carte crée dans un premier temps des cartes dont le microcircuit est vierge de toute information personnalisée, puis dans un deuxième temps, il y inscrit, par programmation des données secrètes, personnalisant chaque carte. Les cartes sont ensuite protégées par des verrous technologiques irréversibles.

Les codes personnels d'identification sont alors envoyés à chaque utilisateur (clients de la banque), et, par une autre route, les cartes elles-mêmes sont expédiées à la banque, où les clients sont priés de venir les retirer. Ce système se révèle d'une grande sécurité contre les tentatives d'usurpation.

Bien entendu, une carte à puce est susceptible de servir à bien d'autres applications, en dehors du domaine bancaire, où l'aspect pratique de portabilité de la carte et de sa sécurité d'utilisation offrent des perspectives intéressantes.

On peut penser à développer un microcircuit spécifique de chaque cas d'application particulier envisagé mais il est plus économique et plus simple de prévoir des microcircuits d'un type suffisamment universel pour que l'une ou l'autre des applications envisagée y soit ensuite programmée.

Dans cette perspective, des problèmes apparaissent déjà du point de vue de la sécurité si on envisage de confier à une entité autre que le fabricant de microcircuit, le soin d'inscrire les données secrètes d'identification et les données fonctionnelles de sa propre application.

En effet une personne mal intentionnée ayant réussi à se procurer des cartes vierges et ayant par ailleurs pris connaissance des techniques d'inscription d'une certaine application, serait susceptible de recréer illicitement des cartes falsifiées en utilisant une technique d'inscription de données qui imite l'originale.

Or, l'entité émettrice de l'application, qui est étrangère au fabricant de microcircuit, peut désirer charger elle-même les données de son application, entre-autres pour ne pas livrer ses secrets au fabricant de microcircuits.

Ces difficultés relatives à la sécurité sont encore aggravées dans le cas où on envisage de créer des cartes dites "multi-application" d'un type universel, vierges à l'origine, et qui sont aptes à être chargées successivement et dans un ordre de succession quelconque, de données et programmes d'applications différents les uns des autres, par des entités émettrices qui sont étrangères les unes aux autres. Le but recherché sera alors d'effectuer une allocation dynamique de la mémoire programmable dans des conditions de sécurité satisfaisantes.

Le fabricant de microcircuits devrait pouvoir assurer une possibilité de protection de plusieurs zones de mémoire dont il ne connaît pas encore les limites individuelles, de manière que chaque entité émettrice puisse protéger la zone où elle a chargé ses données fonctionnelles secrètes, contre toute tentative de lecture ou d'écriture provenant du déroulement d'une autre application, que ces tentatives proviennent d'une erreur de programmation de l'entité ayant émis cette autre application ou qu'elles proviennent de la programmation illicite d'un fraudeur.On doit se rendre compte qu'il serait illusoire de confier à une pluralité d'entités susceptibles de charger diverses applications, des secrets de programmation relatifs au chargement des applications en espérant que ces secrets ne parviennent jamais dans les mains de personnes mal intentionnées. Une solution ne peut être envisagée dans cette direction d'autant plus qu'une entité bien qu'autorisée mais maladroite peut émettre une application contenant une erreur de programmation aboutissant à la destruction d'une donnée contenue dans une application étrangère à la sienne.

L'invention vise tout particulièrement le cas d'un microcircuit pour carte à puce du type multi-applications, dans lequel un haut niveau de sécurité est maintenu malgré que la carte soit fabri-

quée dans un étant vierge et qu'elle soit susceptible d'être chargée, dans un ordre indifférent, par des entités émettrices d'applications qui sont indépendantes les unes des autres. En quelque sorte, le microcircuit souhaité devrait disposer de verrous mobiles mais inviolables que ce soit en cours de chargement, de fonctionnement, ou par l'envoi de signaux quelconques sur les accès d'entrée - sortie.

L'invention à pour but de fournir une solution à ce problème technique.

En effet, selon l'invention, un microcircuit pour carte à puce, d'un type conforme au paragraphe introductif, est caractérisé en ce que, en vue d'un chargement et d'une exploitation de programmes d'applications indépendants les uns des autres, le microcircuit est muni d'une unité centrale de microprocesseur, d'une mémoire à lecture seule (ROM) contenant notamment des routines fonctionnelles exclusivement exécutables par l'unité centrale et sous sa dépendance, en ce que dans l'espace d'adressage de la mémoire programmable sont délimitées une zone dite de répertoire et une zone dite des applications, lesquelles zones sont exclusives l'une de l'autre, en ce que la zone de répertoire contient une table répertoire où les adresses de début et de fin ainsi qu'un code de référence de chacun des programmes d'applications susceptible d'être chargé dans une portion de la zone des applications, sont inscrits au fur et à mesure du chargement desdits programmes et en utilisant une desdites routines fonctionnelles, en ce que le circuit de contrôle d'adressage est agencé pour inhiber les commandes d'écriture et de lecture de la mémoire programmable pour des adresses qui se situent à l'extérieur de l'intervalle défini par les valeurs limites de début et de fin d'une application désignée préalablement sur les accès d'entrée - sortie, limites qui sont mémorisées dans des registres verrouillables, appelés registres de limites tandis que la voie dite d'autorisation prioritaire comporte des moyens pour s'affranchir de l'inhibition précitée en autorisant l'écriture et la lecture de la zone de répertoire à la condition exclusive que ces opérations soient effectuées au moyen d'une desdites routines fonctionnelles durant une étape opératoire prédéterminée durant laquelle les registres contenant les valeurs limites sont déverrouillés.

Ainsi, dans le cas général où un programme d'application est déjà chargé dans la carte et appelé par l'intermédiaire les moyens d'accès entrée - sortie, un protocole standard de reconnaissance de l'utilisation prévu parmi les routines fonctionnelles est exécuté. Après quoi, une autre routine commande une recherche dans la table répertoire, les valeurs limites d'adresse correspondant au programme d'application demandé, identifié dans ladite table par son code de référence, et ces valeurs sont chargées dans les registres de limites, déverrouillés durant cette étape initiale de fonctionnement.

Le programme d'application demandé est ensuite lancé et simultanément, les registres de limites sont verrouillés de sorte que les demandes d'accès en écriture ou en lecture qui désigneraient la mémoire programmable en dehors de la zone d'adresses comprise entre les valeurs limites mémorisées, resteront sans effet, par exemple jusqu'à ce que le microcircuit soit mis hors tension. La voie d'autorisation prioritaire est en effet refermée. De cette manière, un programme d'application est seul à pouvoir lire ou écrire dans la zone d'adresse qui lui est affectée et qui a été fixée lors du chargement. Les autres programmes d'applications sont protégés contre toute tentative de lecture ou d'écriture.

En ce qui concerne le chargement de tout nouveau programme d'applications au sein de la carte, la sécurité provenant de l'absence d'interférence avec d'autres zones de la mémoire programmable est assurée par des moyens similaires.

Toute demande de chargement d'un programme d'application sur les moyens d'accès entrée - sortie fait l'objet d'un protocole standard engagé en liaison avec une routine fonctionnelle spécialisée contenue dans la mémoire à lecture seule. Le code de référence de l'application ainsi que l'espace mémoire requis doivent être annoncés durant ce protocole. Une routine fonctionnelle spécifique a pour effet de rechercher dans la table répertoire quelle est la première adresse disponible dans la zone des applications (l'adresse qui suit l'adresse de fin la plus élevée des programmes d'applications déjà chargés ou la première adresse de la zone des applications lorsque la carte est vierge). L'adresse de fin de l'application demandée est calculée à partir de la demande d'espace mémoire annoncée précédemment. Lorsque cette adresse de fin d'application est compatible avec l'espace mémoire de la zone des applications, les valeurs limites d'adresses de début et de fin ainsi que le code de référence de l'application sont alors inscrits dans la table répertoire et ces valeurs limites sont chargées dans les registres de limites, toujours par le moyen d'une routine fonctionnelle préétablie et intangible.

Durant ces étapes qui précèdent le chargement, les registres de limites sont déverrouillés, et ils peuvent le rester durant le chargement du programme d'application qui s'ensuit. La sécurité recherchée est néanmoins conservée par le fait que le chargement du programme d'application est effectué sous le contrôle de l'unité centrale et par l'effet d'une routine fonctionnelle préétablie et sûre qui exclut toute modification de contenu des re-

gistres de limites.

On peut aussi prévoir un verrouillage de ces registres qui soit commandé par programmation juste avant le chargement.

Toute tentative de chargement en dehors des limites d'adresses établies précédemment restera sans effet. L'opération de chargement peut être complétée, si on le juge nécessaire, par une routine de vérification des données chargées, une vérification de signature, un test de parité...etc... mais de telles opérations n'auront pas pour effet d'accroître de manière significative la sécurité de fonctionnement du microcircuit selon l'invention.

La voie d'autorisation prioritaire peut être commandée de différentes manières en appliquant des conditions logiques appropriées et en fonction du déroulement des étapes opératoires qui suivent la mise sous tension du microcircuit.

Toutefois, selon un mode de mise en oeuvre préféré de l'invention, qui offre toute garantie du point de vue de la sécurité, le microcircuit se caractérise en ce qu'il comporte une bascule bistable, placée dans un premier état exclusivement lors d'une mise sous tension du microcircuit, puis placée dans un deuxième état opposé au premier, dès qu'un compteur ordinal associé à l'unité centrale, contient une valeur d'adresse comprise dans l'espace d'adressage de la mémoire programmable, et en ce que cette bascule bistable délivre un signal de verrouillage/déverrouillage pour la commande de verrouillage des registres de limites et pour la commande de la voie d'autorisation prioritaire.

Du fait qu'il est lié au compteur ordinal, le signal de verrouillage/déverrouillage fournit un moyen inviolable par un programme d'application quelconque pour le verrouillage des registres de limites et de la voie d'autorisation prioritaire, bien que la bascule bistable reste par sa nature un élément réversible dans le temps.

Comme les routines fonctionnelles qui ont pour objet de lire et/ou écrire dans la table répertoire sont inscrites dans la mémoire à lecture seule, le compteur ordinal pointe cette mémoire pour les exécuter pas à pas.

Il suffit donc de placer l'exécution de ces routines dans des étapes opératoires qui précèdent toute exécution de programme inscrit dans la mémoire programmable. La voie d'autorisation prioritaire est encore ouverte tant qu'il s'agit de routines exécutées sous le contrôle de l'unité centrale. Dès que le compteur ordinal sera chargé d'une adresse désignant la mémoire programmable, le signal de verrouillage/déverrouillage sera aussitôt placé dans son état réalisant le blocage, verrouillant les registres de limites et interdisant l'accès à la table répertoire.

En pratique ceci peut avantageusement être réalisé dans un microcircuit caractérisé en ce que le circuit de contrôle d'adressage comporte une porte OU dite de validation, dont la sortie commande la transmission des signaux d'écriture et de lecture de la mémoire programmable, en ce qu'une première entrée de cette porte de validation reçoit un signal résultant de la comparaison de toute adresse appelée avec le contenu des registre de limites, et une autre entrée par laquelle s'effectue l'autorisation prioritaire, reçoit un signal résultant du produit logique entre le signal de verrouillage/déverrouillage et le signal de sortie d'un décodeur d'adresses reconnaissant les seules adresses de la zone de répertoire.

Jusqu'à présent on a considéré que l'espace d'adressage de la mémoire programmable était protégé dans les conditions précitées relatives à la zone de répertoire et la zone des applications.

Il peut cependant être utile de prévoir en outre une certaine zone de l'espace d'adressage de la mémoire programmable qui soit librement accessible par tout programme d'application, par exemple pour y placer provisoirement des résultats de traitement ou pour transmettre des données d'une application à une autre.

Sans amoindrir la sécurité, cette facilité peut être obtenue aisément par une légère modification du circuit de contrôle d'adressage précédemment défini. Selon ce mode de mise en oeuvre, un microcircuit selon l'invention est caractérisé en ce qu'une zone dite "publique" est en outre prévue dans l'espace d'adressage de la mémoire programmable, zone qui est distincte de la zone de répertoire et de la zone des applications, et en ce que le circuit de contrôle d'adressage comporte un décodeur d'adresse supplémentaire, reconnaissant les seules adresses de la zone publique, qui délivre en sortie un signal appliqué à une entrée supplémentaire de la porte OU de validation.

La description qui va suivre en regard des dessins annexés, donnés à titre d'exemples non limitatifs, fera bien comprendre en quoi consiste l'invention et comment elle peut être réalisée.

La figure 1 représente un schéma de principe d'un microcircuit selon l'invention,

la figure 2 est un schéma - bloc d'une portion du microcircuit délivrant un signal de verrouillage/déverrouillage et,

la figure 3 illustre schématiquement l'organisation d'une zone particulière de la mémoire programmable.

La figure 1 représente le schéma général et simplifié d'un microcircuit selon l'invention.

Sur cette figure, et pour plus de clareté, les éléments qui ne concernent pas directement l'invention n'ont pas été représentés. Le microcircuit comporte une unité centrale 10 de microproces-

seur, une mémoire vive 11, une mémoire à lecture seule 12, une mémoire programmable 13 par exemple une EEPROM ainsi qu'un circuit de contrôle d'adressage 14 de la mémoire programmable 13. Les éléments précités sont reliés entre eux par un système de bus 15 véhiculant les adresses et les données entre ces différents éléments. Pour plus simplicité le système de bus 15 a été représenté par une liaison unique bien qu'on puisse, selon les cas, utiliser un bus d'adresses et un bus de données qui soit séparés l'un de l'autre. Comme cela est de pratique courante, le système de bus 15 peut encore être un bus unique sur lequel les adresses et les données sont multiplexées temporellement. A l'unité centrale 10 sont associés des moyens d'accès entrée-sortie 16, moyens qui sont isolés du système de bus 15 de manière à garantir un accès contrôlé par l'unité centrale 10 aux zones sensibles du microcircuit contenant des données à protéger.

Le circuit de contrôle d'adressage 14 reçoit de l'unité centrale 10 un ensemble de liaisons de commande 20 parmi lesquelles figurent essentiellement les commandes d'écriture W de lecture R et d'effacement E lorsqu'elles sont destinées à la mémoire programmable 13.

Les commandes de lecture et d'écriture pour la mémoire vive 11 et les commandes de lecture pour la mémoire à lecture seule 12, n'ont pas été représentées, pour plus de clarté de la figure et du fait qu'elles n'interfèrent pas avec l'invention. Les commandes W, R et E destinées à la mémoire programmable 13 peuvent être inhibées ou validées par le circuit de contrôle d'adressage 14 dans des conditions qui seront expliquées ci-après. Conformément à une pratique en usage dans les microcontrôleurs, la mémoire à lecture seule 12 et la mémoire programmable 13 font partie d'un même système d'adressage tandis que la mémoire vive 11 est adressée de manière distincte et exclusive de l'adressage des mémoires 12 et 13. Dans l'espace d'adressage de la mémoire programmable 13 sont délimitées une zone de répertoire ZR et une zone des applications ZA, zones qui sont exclusives l'une de l'autre et qui sont définies une fois pour toutes par construction.

Initialement les zones ZR et ZA sont vierges. Le microcircuit peut être chargé de programmes d'applications indépendant les uns des autres dans un ordre et à des moments qui sont indifférents. Au fur et à mesure du chargement desdits programmes ceux-ci vont être inscrits successivement dans des zones distinctes ZA_1, ...ZA_i, ...ZA_n qui sont adjacentes et qui remplissent progressivement la zone des applications ZA.

Bien entendu, chaque zone ZA_i d'une application déterminée peut également contenir des données (autres que des codes opératoires) qui sont propres à cette application et/ou des emplacements mémoire destinés à recevoir des résultats de traitement. Selon l'invention la zone de répertoire ZR contient une table répertoire où les adresses de début et de fin ainsi qu'un code de référence de chacun des programmes d'applications chargés dans la zone des applications sont inscrits lors du chargement desdits programmes. Ainsi, un $i^{ème}$ programme d'application chargé dans la zone ZA_i entre les adresses ZA_{il} et ZA_{ih} respectivement de début et de fin du programme d'application i a été chargé au cours d'un protocole au cours duquel les adresses ZA_{il} et ZA_{ih} ont été préalablement déterminées et chargées dans la table répertoire de la zone de répertoire ZR accompagnés d'un code référence du programme d'application i. Le protocole de chargement qui vient d'être mentionné utilise une desdites routines fonctionnelles inscrites de manière immuable dans la mémoire à lecture seule 12.

Le circuit de contrôle d'adressage 14 est agencé pour inhiber les commandes de lecture (R) d'écriture (W) et d'effacement (E) destinées à la mémoire programmable 13 pour des adresses qui se situent à l'extérieur de l'intervalle défini par les valeurs limites de début ZA_{il} et de fin ZA_{ih} d'une application ZA_i désignée préalablement sur les accès d'entrée/sortie 16. Pour ce faire, le système de bus 15 est partagé dès l'entrée dans le circuit de contrôle d'adressage 14 en un bus de données 21 et un bus d'adresses 22. Le bus de données 21 est appliqué en parallèle sur deux registres verrouillables dits registres de limites 25, 26 respectivement, où les adresses de début et de fin d'une application en cours de traitement peuvent être mémorisées et verrouillées au moyen d'un signal de verrouillage/déverrouillage (LOC) provenant d'une borne 60. Le bus d'adresses 22, interne au circuit de contrôle d'adressage 14, est appliqué en parallèle à deux comparateurs 28, 29 respectivement associés aux registres de limites 25 et 26. Le comparateur 28 délivre un signal d'autorisation lorsque l'adresse courante sur le bus d'adresses 22 est supérieure ou égale à l'adresse de début mémorisée dans le registre de limites 25 tandis que le comparateur 29 délivre un deuxième signal d'autorisation lorsque l'adresse courante sur le bus d'adresse 22 est inférieure ou égale à l'adresse limite de fin d'application mémorisée dans le registre 26. Les deux signaux d'autorisation délivrés en sortie des comparateurs 28 et 29 sont appliqués aux deux entrées d'une porte ET 30 dont la sortie 31 présente un signal d'autorisation lorsque l'adresse courante présentée sur le bus d'adresse 22 est comprise dans les limites des adresses mémorisées dans les registres 25 et 26. Le signal d'autorisation présent à la sortie 31 de la porte 30 est appliqué via une porte OU 32, dite de valida-

tion, à l'une des entrées de trois portes ET 35, 36, 37 dont l'autre entrée de chacune d'elle reçoit respectivement les commandes de lecture (R) écriture (W) et d'effacement (E). Ces commandes ne sont donc transmises à la mémoire programmable 13 que lorsque les portes ET 35, 36, 37 sont passantes c'est-à-dire lorsque l'adresse courante présente sur le bus d'adresses 22 est comprise ou égale aux limites d'adresses mémorisées dans les registres 25 et 26 et au contraire ces commandes sont inhibées dans le cas où l'adresse courante est extérieure à ces valeurs limites.

La porte OU 32 constitue une partie de ce qui a été appelé précédemment voie d'autorisation prioritaire, du fait qu'elle comporte une deuxième entrée 40 sur laquelle est appliqué un signal d'autorisation durant une étape opératoire prédéterminée qui suit toute mise sous tension du microcircuit.

Pour réaliser ladite voie d'autorisation prioritaire, le signal de verrouillage/déverrouillage (LOC) est inversé par un inverseur 41 et appliqué à une des deux entrées d'une porte ET 42. Sur l'autre entrée de cette porte ET 42 est appliqué le signal de sortie d'un décodeur d'adresses 43 reconnaissant les seules adresses de la zone de répertoire (ZR) et fournissant à la porte ET 42 un signal d'autorisation lorsque l'adresse courante présentée sur le bus d'adresses 22 est comprise dans l'espace d'adressage de la zone de répertoire (ZR). La porte ET 42 effectue ainsi le produit logique entre le signal de verrouillage/déverrouillage (LOC) et le signal de sortie du décodeur d'adresses 43 pour fournir un signal d'autorisation prioritaire à l'entrée 40 de la porte de validation 32. Le décodeur d'adresses 43 qui reconnaît uniquement des adresses de la zone de répertoire (ZR) de la mémoire programmable 13 peut être d'un type très simple lorsque le nombre d'octets de la zone de répertoire (ZR) peut s'exprimer par une puissance entière de deux telle qu'une valeur de 256 octets ou 512 octets etc. A titre d'exemple, la zone de répertoire peut avoir une adresse de début exprimée en hexadécimal par la valeur 8000 et l'adresse de fin de cette zone par l'adresse hexadécimale 80FF.

Le microcircuit représenté à la figure 1 comporte encore une disposition optionnelle selon laquelle une zone dite "publique" ZB est prévue dans l'espace d'adressage de la mémoire programmable 13, zone qui est distincte de la zone de répertoire ZR et de la zone des applications ZA et que l'on a fait figurer à titre d'exemple entre les zones ZR et ZA et adjacente à celles-ci. Du fait que l'on a prévu un accès inconditionnel à cette zone publique ZP, la porte de validation 32 comporte une troisième entrée 50 par laquelle est appliqué un signal de validation provenant d'un décodeur d'adresses 51 similaire au décodeur d'adresses 43 à l'exception près qu'il ne reconnaît que les seules adresses de la zone publique ZP de la mémoire programmable 13.

Ainsi que cela a déjà été mentionné, le signal de verrouillage/déverrouillage (LOC) est placé dans un état de déverrouillage (LOC = 0) durant une étape opératoire prédéterminée qui suit toute mise sous tension du microcircuit. Il peut être généré par tous systèmes logiques convenables permettant d'autoriser le chargement des registres de limites 25 et 26 et de débloquer la porte ET 42 pour permettre la lecture et/ou l'écriture de la table répertoire située dans la zone de répertoires ZR.

Selon un mode de mise en oeuvre de l'invention particulièrement avantageux pour la sécurité élevée qu'il procure, le signal de verrouillage/déverrouillage (LOC) est produit en liaison avec le contenu d'un compteur ordinal tel qu'il est généralement connu et associé à une unité centrale de microprocesseur.

On se reporte maintenant à la figure 2 pour la description de cette portion du microcircuit.

Sur cette figure, on a représenté un compteur ordinal 55 qui fait partie de l'unité centrale 10 et qui permet l'exécution pas à pas de routines fonctionnelles contenues dans la mémoire à lecture seule 12. Cette disposition est par ailleurs classique dans tous les microcontrôleurs ou microprocesseurs et ne nécessite pas, de ce fait, une description détaillée. Le compteur ordinal 55 est relié à un décodeur 56, 57 qui actionne à sa sortie, via une porte OU 66 une bascule bistable 58, par exemple de type RS, lorsque le contenu du compteur ordinal 55 dépasse une certaine valeur d'adresse qui dans l'exemple illustré a pour limite la valeur hexadécimale 7FFF. Cette limite d'adresse correspond au cas où la mémoire programmable a son adresse la plus basse exprimée en hexadécimal par la valeur 8000. Une routine fonctionnelle, par ailleurs classique en matière de microcontrôleur, effectue la remise à zéro d'un certain nombre de registre et dans le cas précis, applique un signal de remise à zéro de la bascule bistable 58 sur son entrée de remise à zéro 59. La bascule bistable 58 délivre en définitive le signal de verrouillage/déverrouillage (LOC) sur la borne 60 représentée également à la figure 1, à l'entrée du circuit de contrôle d'adressage 14.

A la suite d'une mise sous tension du microcircuit, et tant que l'unité centrale 10 exécute les routines fonctionnelles contenues dans la mémoire à lecture seule 12, le compteur ordinal 55 désigne des adresses de programme qui sont relatives à cette même mémoire à lecture seule et la bascule bistable 58 délivre un signal de déverrouillage (LOC = 0). La table de répertoire peut être lue ou écrite sous le contrôle de l'unité centrale 10 et les

registres de limites 25, 26 peuvent être chargés par cette même unité centrale. Au contraire, dès lors qu'un programme d'application est appelé (de sorte que l'adresse contenue dans le compteur ordinal 55 dépasse nécessairement l'adresse hexadécimale 7FFF, donnée comme limite dans l'exemple) la bascule bistable 58 est placée dans l'état opposé au premier état et délivre un signal de verrouillage sur la borne 60 (LOC = 1). Les registres de limites 25 et 26 ainsi que la voie d'autorisation prioritaire seront désormais verrouillés jusqu'à la prochaine mise hors tension du microcircuit. Comme on le voit cette disposition offre toute sécurité pour garantir une absence d'interférence entre les différents programmes d'applications susceptibles d'être chargés dans la zone des applications ZA.

La porte OU 66 permet également une variante de fonctionnement selon laquelle le signal de verrouillage (LOC = 1) peut être produit par une routine spécifique contenue dans la mémoire ROM 12, et donc même si le contenu du compteur n'a pas encore dépassé la valeur limite de 7FFF. La routine fonctionnelle prévue à cet effet impose à l'unité centrale 10 d'émettre un signal de commande sur une liaison 67 qui est appliquée à une deuxième entrée de la porte OU 66. On peut donc ainsi émettre un signal de verrouillage par programmation qui subsistera jusqu'à la prochaine remise à zéro. Cette disposition permet d'accroître encore la sécurité pendant le chargement d'un nouveau programme d'application et pendant l'exécution d'une application qui ne ferait appel qu'à des routines contenues dans la mémoire ROM 12 et dont la zone ZA_i reservée dans la mémoire programmable ne contiendrait que des données (et aucun code opératoire).

A l'aide de la figure 3, on va expliquer succintement comment la zone de répertoire ZR de la mémoire programmable 13 peut être organisée. La figure 3 se réfère par ailleurs à un exemple dont les valeurs sont purement arbitraires et n'ont pas d'autre but que de rendre les explications plus claires.

On suppose que la zone répertoire de la mémoire programmable 13 commence à l'adresse hexadécimale 8000 et finit à l'adresse 80FF. Dans cette zone d'un volume de 256 octets, on peut prévoir, si on le désire, une portion désignée par ZID destinée à recevoir des données d'identification de la carte, du propriétaire de celle-ci et des clés. En dehors de la zone d'identification ZID se situe la table de répertoire TR proprement dite qui débute à l'adresse INI et s'étend jusqu'à l'adresse hexadécimale 80FF. Comme indiqué sur la figure, la table répertoire TR contient les inscriptions successives d'un premier programme chargé, dont le code de référence est appelé "PR-APP-A", d'une longueur de 350 octets, dont l'adresse hexadécimale de début d'exécution de programme est égale à 8200 ainsi que l'adresse de début de zone, et l'adresse de fin de zone égale à 835D, puis un programme référencé "PR-APP-B" d'une longueur de 250 octets, ayant pour adresse de début d'exécution de programme la valeur hexadécimale 8365, pour l'adresse de début de zone, 835E pour l'adresse de fin 8457, et enfin un programme référencé "PR-APP-C", d'une longueur de 180 octets, ayant pour adresse de début d'exécution de programme la valeur hexadécimale 8460 pour l'adresse de début de zone, 8458 et pour adresse de fin la valeur 850C. En effet, les adresses de début d'exécution de programme, comme l'indique cet exemple, ne coïncident pas nécessairement avec l'adresse de début de zone allouée au même programme, mais elles sont nécessairement comprises dans ladite zone allouée.

On va décrire tout d'abord le cas où un programme d'application déjà chargé dans la carte est appelé sur les moyens d'accès d'entrée/sortie.

Toute demande d'exécution d'un programme d'application fait l'objet d'une procédure qui peut se décomposer en 5 étapes principales :

- une étape d'initialisation qui peut inclure une procédure d'identification de la carte et son utilisateur,
- une procédure de demande de l'application annoncée sur les moyens d'entrée-sortie,
- une procédure de recherche dans la table de répertoire TR pour déterminer si l'application demandée existe, et dans l'affirmative, le chargement des données spécifiques de l'application prélevées dans cette table,
- le chargement des registres de limites,
- et enfin l'exécution proprement dite du programme d'application demandé.

Dès la mise sous tension du microcircuit, la bascule bistable 58 est remise à zéro en même temps qu'un certain nombre de registres et notamment le compteur ordinal 55. Après un éventuel protocole de reconnaissance mettant en jeu des données d'identification contenues dans la portion d'identification ZID de la zone répertoire ZR, un programme d'application par exemple celui référencé "PR-APP-B" est appelé.

Sous l'effet d'une routine préétablie, l'unité centrale 10 effectue une recherche dans la table répertoire TR de manière à déterminer si un tel programme d'application est présent. Dans l'affirmative les valeurs d'adresse de début et de fin de ce programme d'application, dans l'exemple les adresses hexadécimales 835E et 8457 respectivement, sont prélevées dans la table de répertoire TR et toujours sous l'effet d'une routine préétablie ces valeurs limites sont chargées dans les registres des limites 25 et 26 respectivement. Jusqu'à présent le fonc-

tionnement du microcircuit n'a fait appel qu'à des routines situées dans la mémoire à lecture seule 12 de sorte que le compteur ordinal n'a jamais franchi la limite des adresses désignant la mémoire programmable 13 soit dans notre exemple une adresse supérieure à la valeur hexadécimale 7FFF. Dans l'étape opératoire qui suit et qui consiste à exécuter le programme appelé, le compteur ordinal 55 est chargé avec l'adresse de début d'exécution du programme d'application appelé, par exemple l'adresse hexadéciame 8365. Ceci entraîne un changement d'état de la bascule bistable 58 et procure sur la borne 60 un signal de verrouillage (LOC = 1). Ce signal a pour effet de verrouiller les registres de limites 25 et 26 et de bloquer la voie d'autorisation prioritaire formée par la branche comportant l'inverseur 41, la porte ET 42 et l'entrée 40 de la porte de validation 32. Désormais les opérations d'écriture, de lecture et d'effacement de la mémoire programmable ne peuvent plus être exécutées que dans la seule portion comprise dans les limites de début et de fin du programme appelé ou dans la zone publique ZP. Cet accès sélectif va persister jusqu'à la mise hors tension du microcircuit. Les opérations d'écriture et de lecture dans la zone dite publique ZP restent possibles au moyen du signal d'autorisation présent sur l'entrée 50 de la porte de validation 32 et provenant du décodeur d'adresse 51 qui reconnaît les adresses correspondant à cette zone ZP.

On en vient maintenant à décrire le fonctionnement du microcircuit dans le cas où un programme d'application doit être chargé dans la mémoire programmable 13. Dès la mise sous tension du microcircuit, une étape d'initialisation est engagée qui est essentiellement similaire à celle déjà mentionnée pour l'exécution d'un programme d'application. Dans une seconde étape, et sous le contrôle d'une routine fonctionnelle préétablie, le code de référence du programme à charger est annoncé sur les entrées-sorties ainsi que le volume de mémoire nécessaire à ce programme exprimé par exemple en nombre d'octets. Dans une troisième étape opératoire, et également sous l'effet d'une routine fonctionnelle préétablie, une recherche est effectuée dans la table répertoire TR contenue dans la zone de répertoire ZR de manière à déterminer si le programme demandé a déjà été chargé et sinon quelle est la dernière adresse limite de fin de programme inscrite dans la table. Cette adresse correspond également à l'adresse occupée la plus élevée de la zone ZA puisque les programmes ont été chargés successivement à des adresses croissantes. Dans l'exemple il s'agit de la valeur hexadécimale 850C. La même routine fonctionnelle détermine d'une part la valeur d'adresse qui suit immédiatement c'est-à-dire l'adresse hexadécimale 850D qui sera prévue comme valeur limite de début du nouveau programme à charger et calcule d'autre part quelle sera l'adresse de fin de programme à charger compte tenu du volume de mémoire annoncé dans la seconde étape. Dans une quatrième étape, l'unité centrale effectue un test pour déterminer si l'adresse de fin de programme ainsi calculée est compatible avec l'adresse la plus élevée de la zone des applications ZA, puis sous l'effet d'une routine préétablie, les valeurs limites de début et de fin de programme ainsi déterminées sont chargées dans les registres de limites 25 et 26. Du fait que jusqu'à présent l'unité centrale a effectué des routines fonctionnelles situées dans la mémoire à lecture seule 12, le compteur ordinal n'a jamais franchi la limite des valeurs d'adresse qui concernent la mémoire programmable 13. Ainsi le signal de verrouillage/déverrouillage est dans son état de déverrouillage (LOC = 0) ce qui a permis le chargement des registres des limites 25 et 26 ainsi que la lecture de la table de répertoire au moyen de la voie d'autorisation prioritaire 41, 42 et 32. Dans une cinquième étape et sous l'effet d'une routine fonctionnelle, l'unité centrale 10 complète la table répertoire de la zone de répertoire ZR en inscrivant à la suite des références déjà présentes, le code de référence, les valeurs limites d'adresses de début et de fin précédemment déterminées et qui concernent le programme d'application en voie de chargement, ainsi que l'adresse de début d'exécution de ce programme.

Une sixième étape opératoire, qui suit, concerne essentiellement le chargement dudit programme, chargement qui ne peut être accompli qu'à l'intérieur des limites fixées par les valeurs d'adresses de début et de fin mémorisées dans les registres des limites 25 et 26. Pendant ce chargement les registres de limites 25 et 26 peuvent ne pas être verrouillés (LOC = 0) mais lesdits registres effectuent le contrôle des opérations d'écriture de la même manière que s'ils avaient été verrouillés. La sécurité est néanmoins maintenue du fait que le chargement est exécuté sous le contrôle de l'unité centrale 10 par l'effet d'une routine fonctionnelle qui ne peut être modifiée par un utilisateur. Toutefois, il est également possible d'activer la bascule 58 par programmation au moyen d'un signal transmis par la liaison 67 à la porte OU 66. Dans ce cas les registres de limites peuvent être verrouillés lors du chargement. Toute tentative de chargement d'un programme d'application d'une longueur supérieure au volume de mémoire annoncé restera inefficace par l'effet du contrôle exercé par les registres de limites 25 et 26 et de la porte ET 30.

Si on le désire l'étape de chargement du programme peut être suivie par une vérification des données chargées, une vérification de signature

finale, un test de parité, et.... Il est aisé de détecter une tentative de chargement d'un programme de longueur supérieure à la longueur annoncée du fait que la ou les dernières données présentées au chargement ne sont pas en fait enregistrées dans la mémoire programmable de sorte qu'une vérification de ces données finales révèle la tentative d'un chargement abusif qu'il soit accidentel ou frauduleux.

En se référant à nouveau à la figure 1, un dispositif additionnel pour la signalisation d'une anomalie de fonctionnement va maintenant être décrit. Ce dispositif additionnel comporte une porte ET 62 à deux entrées, une porte OU 63 à 3 entrées et un inverseur 64.

L'inverseur 64 reçoit en entrée le signal de sortie de la porte de validation 32, et le transmet après inversion, à une entrée de la porte ET 62.

La porte OU 63 a sa sortie connectée à l'autre entrée de la porte ET 62 et reçoit en entrée les trois signaux de commande R, W, E destinés à la mémoire programmable 13, en formant ainsi la somme logique de ces signaux de commande.

Lorsque l'une de ces opérations est demandée, le signal en sortie de la porte OU 63 est à l'état haut. Simultanément, si aucune des conditions d'autorisation n'est réalisée, ce qui produit un signal à l'état bas en sortie de la porte OU 32, la porte ET 62 est alors activée et produit en sortie un signal renvoyé à l'unité centrale 10, par exemple un signal d'interruption (INT = 1).

Ainsi, dans le cas d'utilisation d'une interruption, une anomalie de fonctionnement et plus particulièrement une demande non autorisée d'accès à la mémoire programmable 13 aboutira à une interruption de fonctionnement du microcircuit (interruption non masquable) et on peut, si on le désire, produire un message d'avertissement sur le terminal d'exploitation de la carte.

Le signal INT qui est produit en cas d'anomalie peut être utilisé pour opérer toute modification désirable du fonctionnement du microcircuit : par exemple invalider le programme qui a produit l'anomalie, voire invalider totalement le fonctionnement de la carte.

Une conséquence avantageuse de l'invention est que seules les routines fonctionnelles incrites dans la mémoire à lecture seule devront être extensivement testées et approuvées alors que les programmes d'applications pourront être créés par diverses entités, sous leur propre responsabilité, et sans risque d'interférence indue entre les applications.

Le microcircuit selon l'invention offre donc une parfaite sécurité pour une utilisation de carte à puce multi - applications.

Revendications

1. Microcircuit pour carte à puce comprenant entre - autres : des moyens d'accès entrée - sortie, au moins une mémoire qui est programmable et accessible par un espace d'adressage à adresses consécutives, un circuit de contrôle d'adressage de cette mémoire programmable déterminant soit une inhibition, soit une validation des commandes d'écriture et/ou de lecture par la comparaison de l'adresse demandée avec deux valeurs limites de début et de fin d'une zone particulière d'adresses, limites qui sont mémorisées au sein du circuit de contrôle d'adressage lequel circuit comporte également une voie d'autorisation prioritaire permettant de s'affranchir temporairement de ladite inhibition dans des circonstances prédéterminées et particulières, caractérisé en ce que, en vue d'un chargement et d'une exploitation de programmes d'applications indépendants les uns des autres, le microcircuit est muni d'une unité centrale de microprocesseur, d'une mémoire à lecture seule (ROM) contenant notamment des routines fonctionnelles exclusivement exécutables par l'unité centrale et sous sa dépendance, en ce que dans l'espace d'adressage de la mémoire programmable sont délimitées une zone dite de répertoire et une zone dite des applications, lesquelles zones sont exclusives l'une de l'autre, en ce que la zone de répertoire contient une table répertoire où les adresses de début et de fin ainsi qu'un code de référence de chacun des programmes d'applications susceptible d'être chargé dans une portion de la zone des applications, sont inscrits au fur et à mesure du chargement desdits programmes et en utilisant une desdites routines fonctionnelles, en ce que le circuit de contrôle d'adressage est agencé pour inhiber les commandes d'écriture et de lecture de la mémoire programmable pour des adresses qui se situent à l'extérieur de l'intervalle défini par les valeurs limites de début et de fin d'une application désignée préalablement sur les accès d'entrée - sortie, limites qui sont mémorisées dans des registres verrouillables, appelés registres de limites, tandis que la voie dite d'autorisation prioritaire comporte des moyens pour s'affranchir de l'inhibition précitée en autorisant l'écriture et la lecture de la zone de répertoire à la condition exclusive que ces opérations soient effectuées au moyen d'une desdites routines fonctionnelles durant une étape opératoire prédéterminée durant laquelle les registres contenant les valeurs limites sont déverrouillés.

2. Microcircuit selon la revendication 1, caractérisé en ce qu'il comporte une bascule bistable, placée dans un premier état exclusivement lors d'une mise sous tension du microcircuit, puis placée dans un deuxième état opposé au premier dès qu'un compteur ordinal associé à l'unité centrale, contient une valeur d'adresse comprise dans l'espace d'adressage de la mémoire programmable, et en ce que cette bascule bistable délivre un signal de verrouillage/déverrouillage pour la commande de verrouillage des registres de limites et pour la commande de la voie d'autorisation prioritaire.

3. Microcircuit selon la revendication 2, caractérisé en ce que ladite bascule bistable peut également être placée dans ledit deuxième état par un signal de commande délivré par l'unité centrale sous l'effet d'une routine fonctionnelle prévue à cet effet.

4. Microcircuit selon l'une des revendications 1 à 3, caractérisé en ce que le circuit de contrôle d'adressage comporte une porte OU dite de validation, dont la sortie commande la transmission des signaux d'écriture et de lecture de la mémoire programmable, en ce qu'une première entrée de cette porte de validation reçoit un signal résultant de la comparaison de toute adresse appelée avec le contenu des registres de limites, et une autre entrée par laquelle s'effectue l'autorisation prioritaire, reçoit un signal résultant du produit logique entre le signal de verrouillage/déverrouillage et le signal de sortie d'un décodeur d'adresses reconnaissant les seules adresses de la zone de répertoire.

5. Microcircuit selon la revendication 4, caractérisé en ce qu'une zone dite "publique" est en outre prévue dans l'espace d'adressage de la mémoire programmable, zone qui est distincte de la zone de répertoire et de la zone des application, et en ce que le circuit de contrôle d'adressage comporte un décodeur d'adresse supplémentaire reconnaissant les seules adresses de la zone publique, qui délivre en sortie un signal appliqué à une entrée supplémentaire de la porte OU de validation.

6. Microcircuit selon l'une des revendications 4 ou 5, caractérisé en ce qu'un signal de détection d'anomalie (INT) est produit en sortie d'une porte ET additionnelle dont une première entrée reçoit le signal issu de la porte de validation, et une deuxième entrée reçoit la somme logique des signaux de commande de lecture, d'écriture et d'effacement, à la sortie d'une porte OU additionnelle.

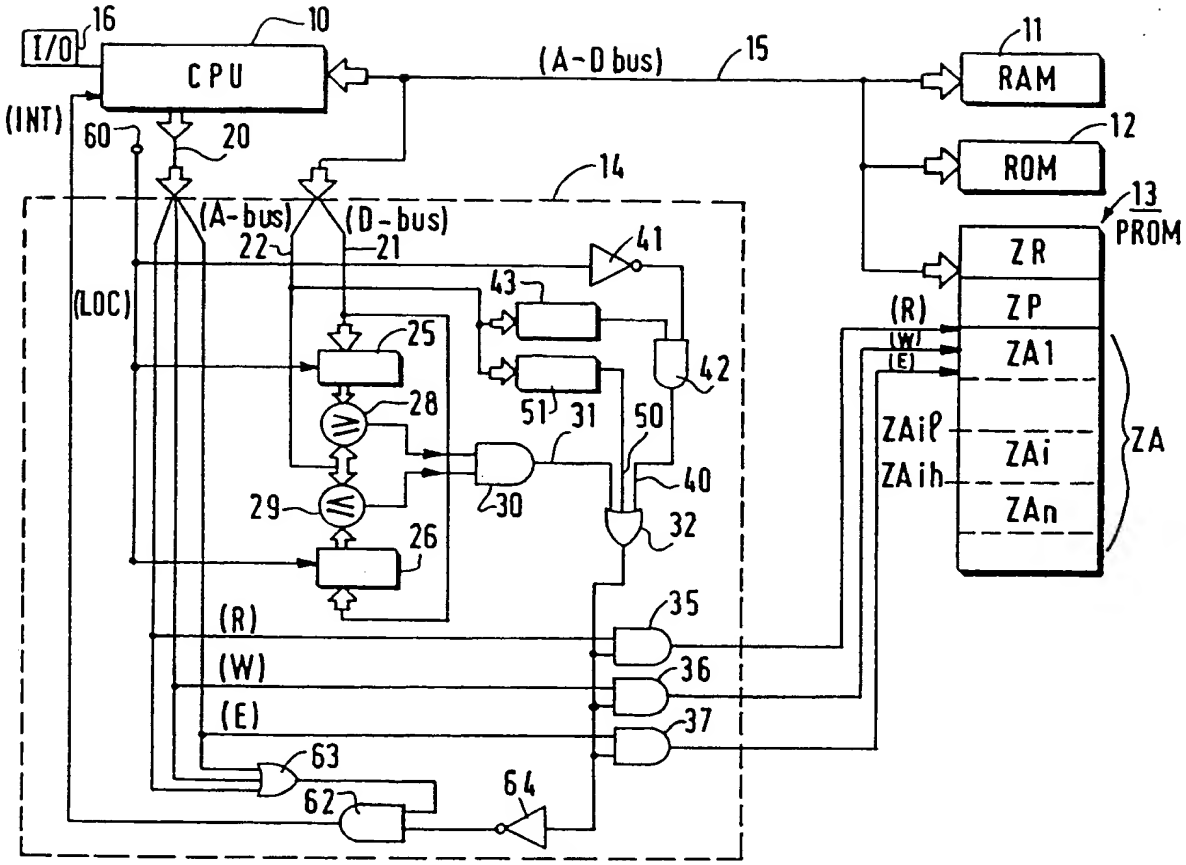


FIG.1

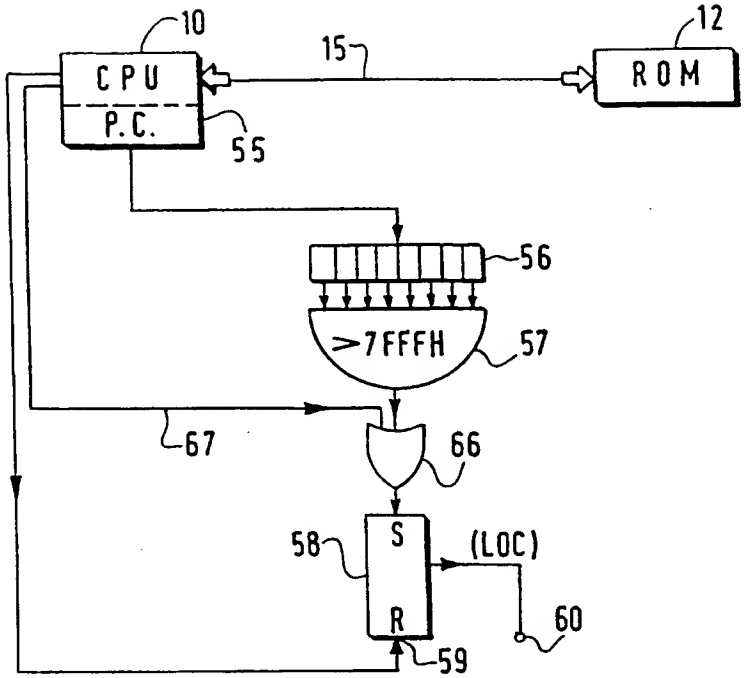


FIG.2

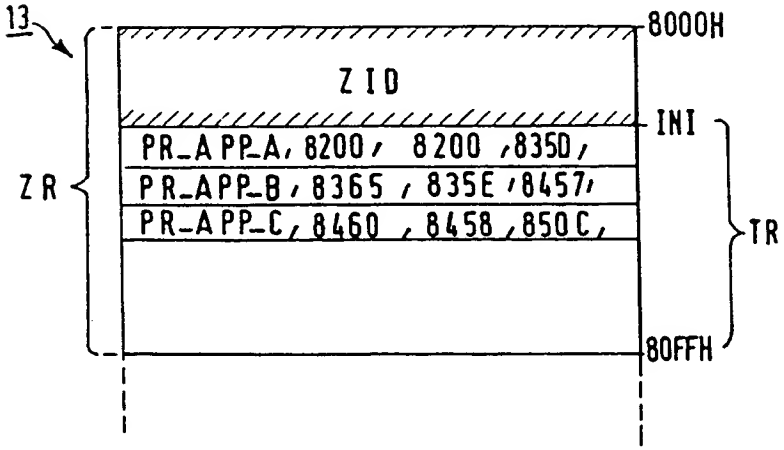


FIG.3

Office européen des brevets

RAPPORT DE RECHERCHE EUROPEENNE

Numero de la demande

EP 92 20 3251

DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	Revendication concernée	CLASSEMENT DE LA DEMANDE (Int. Cl.5)
A	FR-A-2 633 749 (MITSUBISHI) * page 7, ligne 7 - page 8, ligne 17; revendication 1; figures 2,3 *	1	G06K19/073 G07F7/10 G06F12/14
A	EP-A-0 331 407 (HITACHI) * revendications 1,6; figure 1 *	1	
A	FR-A-2 621 409 (THOMSON) * page 3, ligne 4 - page 4, ligne 17 *	1	
A	EP-A-0 451 936 (HITACHI) * abrégé * * colonne 1, ligne 48 - colonne 2, ligne 34 * * colonne 4, ligne 22 - ligne 51; figures 1-3 *	1	
D,A	FR-A-2 304 989 (SOCIETE INTERNATIONALE POUR L' INNOVATION) * revendication 1 *	1	
A	FR-A-2 640 783 (HITACHI) * page 8, ligne 28 - page 9, ligne 14; figures 1,6 *	1	DOMAINES TECHNIQUES RECHERCHES (Int. Cl.5) G06K G07F
A	US-A-3 742 458 (T.INOUE ET AL.) * abrégé *	1	G06F

Le présent rapport a été établi pour toutes les revendications

Lieu de la recherche	Date d'achèvement de la recherche	Examinateur
BERLIN	27 NOVEMBRE 1992	DUCREAU F.

CATEGORIE DES DOCUMENTS CITES

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : arrière-plan technologique
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet antérieur, mais publié à la date de dépôt ou après cette date
D : cité dans la demande
L : cité pour d'autres raisons

& : membre de la même famille, document correspondant

EPO FORM 1503 03.82 (P0402)

(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

PARIS

(11) N° de publication : **2 759 795**
(à n'utiliser que pour les commandes de reproduction)

(21) N° d'enregistrement national : **97 01735**

(51) Int Cl[6] : **G 06 F 12/06**, G 11 C 16/02, G 06 K 19/07

(12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt** : 14.02.97.

(30) **Priorité** :

(43) **Date de mise à la disposition du public de la demande** : 21.08.98 Bulletin 98/34.

(56) **Liste des documents cités dans le rapport de recherche préliminaire** : *Se reporter à la fin du présent fascicule*

(60) **Références à d'autres documents nationaux apparentés** :

(71) **Demandeur(s)** : *FRANCOIS CHARLES OBERTHUR FIDUCIAIRE* — FR.

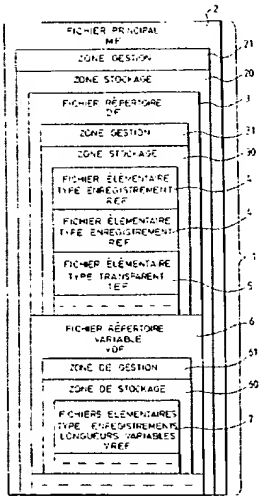
(72) **Inventeur(s)** : DEVAUX FRANCOIS et PERROT DANIEL.

(73) **Titulaire(s)** :

(74) **Mandataire(s)** : THOMSON CSF.

(54) PROCEDE DE STOCKAGE DE DONNEES DANS UNE MEMOIRE REINSCRIPTIBLE DE CARTE A PUCE.

(57) Ce procédé de stockage consiste à ajouter, aux fichiers élémentaires de longueurs fixes de types transparent (5) ou d'enregistrement (4) prévus par la norme ISO 7816-4, des fichiers élémentaires de longueurs variables (7) dont la capacité est toujours adaptée à la taille des enregistrements qu'ils stockent, et à gérer ce nouveau type de fichier avec des micro-instructions respectant le formalisme de la norme ISO 7816-4 et appartenant à la classe des instructions propriétaire. Grâce à ce nouveau type de fichier élémentaire de longueur variable, il est possible d'envisager de nouvelles cartes à puce respectant la norme ISO 7816-4 et mettant en œuvre des techniques de compression de données au niveau du stockage des données, ce qui est particulièrement intéressant compte tenu des capacités de stockage de données limitées d'une carte à puce.



FR 2 759 795 - A1

La présente invention concerne le stockage de données dans une carte à puce à microprocesseur.

Les cartes à puce sont généralement des cartes du format d'une carte de crédit ou des jetons munis d'un microcircuit électronique, à base de mémoires et d'un microcontrôleur, agencés pour permettre le déroulement d'une transaction, par exemple bancaire ou de santé. Elles communiquent avec leur environnement au moyen de lecteurs avec lesquels elles échangent des messages et répondent à une norme ISO 7816-4. Pour assurer le déroulement d'une transaction, elles ont besoin de conserver et de mettre à jour un certain nombre d'informations dans une mémoire embarquée reprogrammable dite EEPROM (Electrically Erasable Programmable Read Only Memory en langue anglo-saxonne).

Jusqu'à présent, le stockage de données dans l'EEPROM d'une carte à puce est organisé par la norme ISO 7816-4 en trois niveaux de fichiers :

- un premier niveau dit fichier principal (Main File en langue anglo-saxonne) constitué de la partie accessible de l'espace de la mémoire EEPROM, pourvu d'un en-tête de définition et d'en-têtes de repérage des fichiers de deuxième niveau qu'il contient;

- un deuxième niveau de fichiers répertoire (Dedicated File en langue anglo-saxonne) pourvus chacun d'un en-tête de définition et d'en-têtes de repérage des fichiers de troisième niveau qu'il contient, et

- un troisième niveau de fichiers dit élémentaires (Elementary File en langue anglo-saxonne) qui sont de deux types : soit de type transparent (Transparent File en langue anglo-saxonne), les données des enregistrements n'étant pas structurées au sein du fichier, l'adressage en écriture et lecture étant laissé à la charge du programme applicatif contrôlant la transaction se déroulant dans la carte à puce, soit de type enregistrement (Record File en langue anglo-saxonne), les données étant stockées par blocs de tailles fixes gérés par le système d'exploitation (Operating System en langue anglo-saxonne) du microcontroleur de la carte à puce.

Avec un système de stockage de données en mémoire EEPROM de carte à puce tel qu'il est régi par la norme ISO 7816-4, le

programme applicatif contrôlant la transaction se déroulant dans la carte à puce doit connaître les tailles des blocs de données qu'il se propose de mémoriser dans la mémoire EEPROM de la carte à puce car les deux types de fichiers élémentaires admis, où sont effectivement stockées les données, sont de longueurs fixes. Cette limitation empêche de recourir à la compression de données au sein de la carte à puce, car, dans ce cas, le programme applicatif contrôlant la transaction se déroulant dans la carte à puce ne maîtrise plus la longueur des blocs de données après compression qui seront effectivement mémorisés.

La présente invention a pour but d'éviter cette limitation en créant un nouveau type de fichier de troisième niveau dit fichier élémentaire de longueur variable (Record Variable File en langue anglo-saxonne) géré par des instructions propriétaires de la norme ISO 7816-4, cela pour garder une compatibilité ascendante avec cette norme.

Elle a pour objet un procédé de stockage de données dans une mémoire réinscriptible de carte à puce consistant à stocker les données au moins en partie, dans des fichiers élémentaires de longueurs variables composés chacun d'une chaîne de longueur variable de domaines de longueurs fixes et de faibles capacités individuelles de la mémoire réinscriptible, ladite chaîne ayant ses domaines repérés au moyen d'une table d'allocation de domaines évoluant en fonction du nombre de données effectivement stockées.

Avantageusement, lesdits fichiers élémentaires d'enregistrements de longueurs variables font partie de fichiers répertoire plus grands, de type variable, contenant leurs tables d'allocation de domaines qui sont constituées chacune d'un en-tête placé dans une zone de gestion du fichier répertoire hôte et localisant le début de chaîne, et de liens placés en début ou en fin de chaque domaine identifiant l'appartenance du domaine concerné à une chaîne, c'est-à-dire son occupation, et localisant le domaine suivant dans la chaîne.

Avantageusement, lesdits fichiers élémentaires de longueurs variables cohabitent avec des fichiers élémentaires de longueurs fixes au sein de fichiers répertoire distincts et sont gérés par un système d'exploitation répondant à des commandes constituées de plusieurs

champs successifs : un champ de classes d'instructions, un champs d'instructions et un champ de paramètres, la classe d'instructions permettant de distinguer un fichier répertoire de type variable contenant des fichiers de longueurs variables, des fichiers répertoire de type fixe contenant des fichiers de longueurs fixes afin que chaque fichier élémentaire soit géré selon son genre, par le système d'exploitation soit seul, soit sous la dépendance d'un programme applicatif intervenant à un niveau supérieur.

D'autres caractéristiques et avantages de l'invention ressortiront de la description ci-après d'un mode de réalisation donné à titre d'exemple. Cette description sera faite en regard du dessin dans lequel :

- une figure 1 illustre de manière schématique un plan d'occupation d'une mémoire morte réinscriptible de carte à puce tel qu'il résulte d'un procédé de stockage de données sur carte à puce selon l'invention, compatible de manière ascendante avec la norme ISO 7816-4, et

- une figure 2 détaille la constitution d'un fichier répertoire de type variable apparaissant dans la figure 1.

Selon la norme ISO 7816-4, le stockage de données dans une mémoire réinscriptible de type EEPROM de carte à puce se fait à l'aide d'un système de fichiers organisé en trois niveaux :

- un premier niveau constitué d'un fichier principal "Main File MF" 2 occupant toute la partie accessible 1 de l'espace mémoire de la mémoire EEPROM avec une zone de stockage 20 complétée par une zone de gestion 21,

- un deuxième niveau de fichiers répertoire "Dedicated file DF" 3 avec également une zone de stockage 30 complétée par une zone de gestion 31. Ces fichiers répertoire DF 3 occupent la zone de stockage 20 du fichier principal MF 2 avec, dans la zone de gestion 21 du fichier principal 2, des en-têtes répertoriant et situant les fichiers répertoire DF 3 au sein du fichier principal 2 (adresses des débuts de fichiers répertoire, tailles des fichiers de répertoire, etc.), et

- un troisième niveau de fichiers élémentaires "Elementary File EF" 4, 5 occupant les zones de stockage 30 des fichiers répertoire DF 3

avec, dans la zone de gestion 31 des fichiers répertoire DF 3 qui les hébergent, des en-têtes répertoriant et situant les fichiers élémentaires EF 4, 5 au sein des fichiers répertoire DF 3 (types des fichiers élémentaires, adresses des débuts des fichiers élémentaires EF 4, 5, tailles des fichiers élémentaires, etc.).

Toujours selon la norme ISO 7816-4, les fichiers élémentaires sont de deux types : les fichiers élémentaires de tailles fixes dits d'enregistrement "Record EF" 4 dans lesquels les données sont stockées par blocs de tailles fixes et les fichiers élémentaires dit transparents "Transparent EF" 5 également de tailles fixes, dans lesquels les données ne sont pas structurées, l'adressage relatif en écriture et lecture étant laissé à la charge du programme applicatif contrôlant le déroulement d'une transaction dans la carte à puce.

Avec un tel système de stockage ne comportant que des fichiers élémentaires de type d'enregistrement 4 ou transparent 5, le programme applicatif contrôlant le déroulement d'une transaction à l'aide d'une carte à puce doit connaître la longueur des blocs de données à stocker dans la carte. Cela ôte tout intérêt aux traitements de compression de données mis en oeuvre au niveau inférieur du système d'exploitation car le programme applicatif contrôlant le déroulement d'une transaction ignore le résultat d'une compression au niveau du système d'exploitation et ne peut en tenir compte pour économiser de la place lors des inscriptions en mémoire dans la carte à puce. Pourtant, la compression de données au niveau du système d'exploitation semble particulièrement indiquée pour une carte à puce en raison des limitations des capacités de stockage de données de cette dernière. Pour remédier à cette limitation, on propose de créer un nouveau type de fichier élémentaire de longueur variable et de l'ajouter aux types de fichiers élémentaires existants de longueurs fixes tout en continuant à respecter la norme ISO 7816-4 pour maintenir une compatibilité ascendante entre cartes à puce.

Le nouveau type de fichier élémentaire de longueur variable VREF 7 est basé sur la constitution d'une chaîne de longueur variable de domaines élémentaires 8 de faible capacité unitaire et de longueurs fixes se partageant la zone de stockage 60 d'un fichier répertoire 6 d'un

nouveau type dit fichier répertoire variable VDF 6 créé pour l'occasion, dont la structure est détaillée à la figure 2.

Ce fichier répertoire variable VDF 6 est distingué des fichiers répertoire classiques DF 3 par un identificateur spécifique inscrit dans l'en-tête qui lui est réservé dans la zone de gestion 21 du fichier principal MF 2. Il présente, comme les fichiers répertoire classiques DF 3 une zone de stockage 60 complétée par une zone de gestion 61.

La zone de stockage 60 d'un fichier répertoire variable VDF 6 est partagée en domaines élémentaires 8 de même longueur et de faible capacité et peut contenir un nombre variable de fichiers de longueurs variables VREF 7. Ces domaines élémentaires, par exemple des multiples de 16 octets, présentent, en début ou en fin, une plage de un ou deux octets exclue du stockage de données et réservée à des liens destinés au chaînage. Ces liens donnent l'état d'occupation ou de non-occupation de chaque domaine élémentaire ainsi que l'adresse du domaine élémentaire suivant, lorsque le domaine élémentaire considéré fait partie d'une chaîne formant un fichier élémentaire de longueur variable VREF 7 et qu'il n'est pas le dernier de la chaîne, le domaine élémentaire suivant n'étant pas nécessairement contiguë. Les liens d'un domaine élémentaire sont constitués, par exemple, par un nombre prenant la valeur 0 pour signifier la non-occupation du domaine considéré, la valeur d'une adresse de domaine repérant le prochain domaine avec lequel le domaine considéré est chaîné, ou une valeur particulière supérieure aux adresses des domaines signifiant la fin d'une chaîne.

La zone de gestion 61 d'un fichier répertoire variable VDF 6 contient des en-têtes 610, 611 de fichiers élémentaires de longueurs variables VREF 7 renfermant principalement l'adresse de début de la chaîne de domaines élémentaires 8 de la zone de stockage 60 affectée au fichier élémentaire de longueur variable considéré.

La création d'un fichier répertoire variable VDF 6 se manifeste par :

- l'écriture dans la zone de gestion 21 du fichier principal MF 2 d'un en-tête affecté à ce fichier répertoire variable, l'identifiant et lui réservant un certain emplacement dans la zone de stockage 20 du

fichier principal MF 2. Cet entête est celui décrit par la norme ISO 7816-4 mais son type devient type DF variable,

- le partage de l'emplacement réservé en une zone de gestion 61 et une zone de stockage 60,

- l'inscription dans la nouvelle zone de gestion crée 61 d'un certain nombre de caractéristiques du nouveau fichier répertoire variable dont la taille du répertoire des en-têtes de fichiers élémentaires de longueurs variables, un indicateur global d'effacement, la taille de ses domaines élémentaires, l'adresse du premier domaine élémentaire, etc., et

- le formatage en domaines élémentaires de la nouvelle zone de stockage 60 avec initialisation des liens.

Une fois qu'un fichier répertoire variable VDF 6 existe, la création d'un fichier élémentaire de longueur variable se manifeste par :

- l'identification du fichier par affectation d'un en-tête dans la zone de gestion 61 du fichier répertoire variable VDF 6 considéré, avec un identifiant "id-file" normalisé ISO 7816-4 et le repérage du premier domaine élémentaire libre trouvé dans la zone de stockage en vue de lui être affecté, le positionnement d'un indicateur d'effacement et

- l'écriture des données affectées au fichier en commençant par le premier domaine élémentaire libre repéré puis en débordant sur d'autres domaines libres jusqu'à épuisement des données à inscrire, avec, à chaque saut de domaine élémentaire, l'inscription dans les liens du domaine élémentaire que l'on vient de quitter de l'adresse du suivant et la mise à jour dans les liens du domaine élémentaire suivant de l'identificateur d'occupation.

On crée ainsi un fichier élémentaire de longueur variable de longueur adaptée à chaque fois, à la quantité précise de données à stocker par une mise en chaîne plus ou moins longue de domaines élémentaires de faibles capacités.

Dans le cas d'un fichier classique REF ou TEF de longueur fixe, le nombre d'enregistrements, c'est-à-dire d'écritures partielles du fichier est fixé une fois pour toutes, la création du fichier impliquant la réservation d'une capacité mémoire non nécessairement utilisée immédiatement qui devient indisponible pour la création d'autres fichiers.

Ce n'est plus le cas avec un fichier de longueur variable VREF qui n'occupe que la capacité mémoire strictement nécessaire aux enregistrements qu'il renferme et qui laisse libre la mémoire inutilisée pour d'autres fichiers.

La gestion des fichiers élémentaires transparents TEF 5, d'enregistrements classiques REF 4 ou à longueurs variables VREF 7, c'est-à-dire leurs créations, inscriptions, lectures, effacements, suppressions, se fait, de manière classique au moyen de routines du système d'exploitation du microcontrôleur de la carte à puce appelées par l'intermédiaire de l'interpréteur de commandes du système d'exploitation au moyen de commandes respectant le formalisme de la norme ISO 7816-4, c'est-à-dire avec une écriture composée de plusieurs champs successifs : un champ de classes d'instructions, un champ d'instructions et un champ de paramètres. De préférence, on utilise les commandes habituelles de la norme ISO 7816-4 pour la gestion des fichiers élémentaires transparents TEF et d'enregistrement REF, et les mêmes commandes avec un champ de classe d'instructions propriétaire pour les fichiers élémentaires de longueurs variables. Grâce à cela, on obtient une compatibilité ascendante avec les cartes à puce respectant la norme ISO 7816-4 et ne connaissant que les deux types habituels de fichiers élémentaires que sont les fichiers élémentaires transparents TEF et les fichiers élémentaires d'enregistrement REF.

Bien évidemment, l'écriture ou l'effacement d'un fichier élémentaire de longueur variable respecte les règles de sécurité habituelles.

Une création de fichier ou une écriture n'est réalisée que s'il y a la place nécessaire à l'action à faire. Une écriture commencée doit se terminer pour être validée. Cela s'obtient, conformément à la norme 7816-4 au moyen d'un bit auxiliaire d'activité placé dans l'en-tête concernant le fichier, mis à un avant l'action envisagée et remis à zéro après la fin de l'action. Si le bit d'activité est maintenu à un après une action, le fichier est déclaré invalide et l'on ne peut plus y accéder. Enfin une taille maximale d'enregistrement est prévue dans la zone de gestion du fichier répertoire variable hôte VDF de manière à surveiller les transferts de données.

De la même façon, un effacement commencé doit être achevé pour être validé. Cela s'obtient au moyen d'un bit d'effacement placé dans l'en-tête concernant le fichier, mis à un avant l'effacement qui consiste à réinitialiser les liens des domaines élémentaires appartenant à la chaîne constituant le fichier, puis remis à zéro en fin d'action juste avant la suppression de l'en-tête de fichier. Si une action d'effacement de fichier est interrompue par retrait inopportun de la carte à puce de son lecteur, le bit d'effacement le signale et permet l'achèvement de l'action interrompue d'effacement en préalable à toute utilisation ultérieure de la carte à puce.

Bien évidement, l'invention n'est pas limitée au mode de réalisation décrit mais s'étend à toutes les variantes qui sont à la portée de l'homme du métier. Les liens des domaines élémentaires qui constituent une sorte de table d'allocation des domaines peuvent être déportés en dehors de la zone de stockage du fichier répertoire de type variable hôte et réunis dans une plage de la zone de gestion de ce dernier.

REVENDICATIONS

1. Procédé de stockage de données dans une mémoire réinscriptible de carte à puce caractérisé en ce qu'il consiste à stocker les données au moins en partie, dans des fichiers élémentaires de longueurs variables (7) composés chacun d'une chaîne de longueur variable de domaines de longueurs fixes (8) et de faibles capacités individuelles de la mémoire réinscriptible, ladite chaîne ayant ses domaines (8) repérés au moyen d'une table d'allocation de domaines évoluant en fonction du nombre des données effectivement stockées.

2. Procédé selon la revendication 1, caractérisé en ce que les fichiers élémentaires de longueurs variables (7) sont localisés dans la mémoire réinscriptible au sein de fichiers répertoire plus grands, de type variable (6) contenant leurs domaines (8) et leurs tables d'allocation de domaines.

3. Procédé selon la revendication 2, caractérisé en ce qu'une table d'allocation de domaines comporte un en-tête (610) placé dans une zone de gestion (61) du fichier répertoire variable (6) hôte localisant l'adresse du domaine (8) de début de chaîne dans le fichier répertoire variable (6) hôte, et des liens placés dans des emplacements réservés dans chaque domaine du fichier répertoire variable (6) hôte identifiant l'appartenance de chaque domaine à une chaîne.

4. Procédé selon la revendication 1, caractérisé en ce que lesdits domaines (8) ont une capacité égale ou multiple de 16 octets.

5. Procédé selon la revendication 3, caractérisé en ce que lesdits liens occupent entre un et deux octets dans chaque domaine (8).

6. Procédé selon la revendication 3, caractérisé en ce que lesdits liens d'un domaine (8) renferment un nombre prenant la valeur 0 pour signifier la non occupation du domaine considéré, la valeur d'une adresse de domaine repérant le prochain domaine avec lequel le domaine

considéré est chaîné, ou une valeur particulière supérieure aux adresses des domaines signifiant la fin d'une chaîne.

7. Procédé selon la revendication 1, caractérisé en ce que la gestion des dits fichiers élémentaires de longueurs variables se fait au moyen de routines d'un système d'exploitation appelées par l'intermédiaire de commandes spécifiques comprises par l'interpréteur de commandes dudit système d'exploitation et reprenant dans leurs formulations un champ de classe d'instructions, un champ d'instructions et un champ de paramètres.

8. Procédé selon la revendication 1, caractérisé en ce qu'il consiste à stocker les données en partie, dans des fichiers élémentaires de longueurs variables (7) composés chacun d'une chaîne de longueur variable de domaines de longueurs fixes (8) et de faibles capacités de la mémoire réinscriptible, ladite chaîne ayant ses domaines (8) repérés au moyen d'une table d'allocation de domaines évoluant en fonction des variations du nombre de données effectivement stockées, et en partie dans des fichiers élémentaires de longueurs fixes (4, 5).

9. Procédé selon la revendication 8, caractérisé en ce que les fichiers élémentaires de longueurs variables (7) et les fichiers élémentaires de longueurs fixes (4, 5) sont localisés dans la mémoire réinscriptible dans des fichiers répertoire distincts (3, 6) de plus grandes capacités, des fichiers répertoire variables (6) contenant les domaines (8) et les tables d'allocation de domaines des fichiers élémentaires de longueurs variables et des fichiers répertoire fixes (3) contenant les fichiers élémentaires de longueurs fixes (4, 5) et des en-têtes identifiant les fichiers élémentaires de longueurs fixes par les adresses de leurs débuts et par les mentions de leurs capacités.

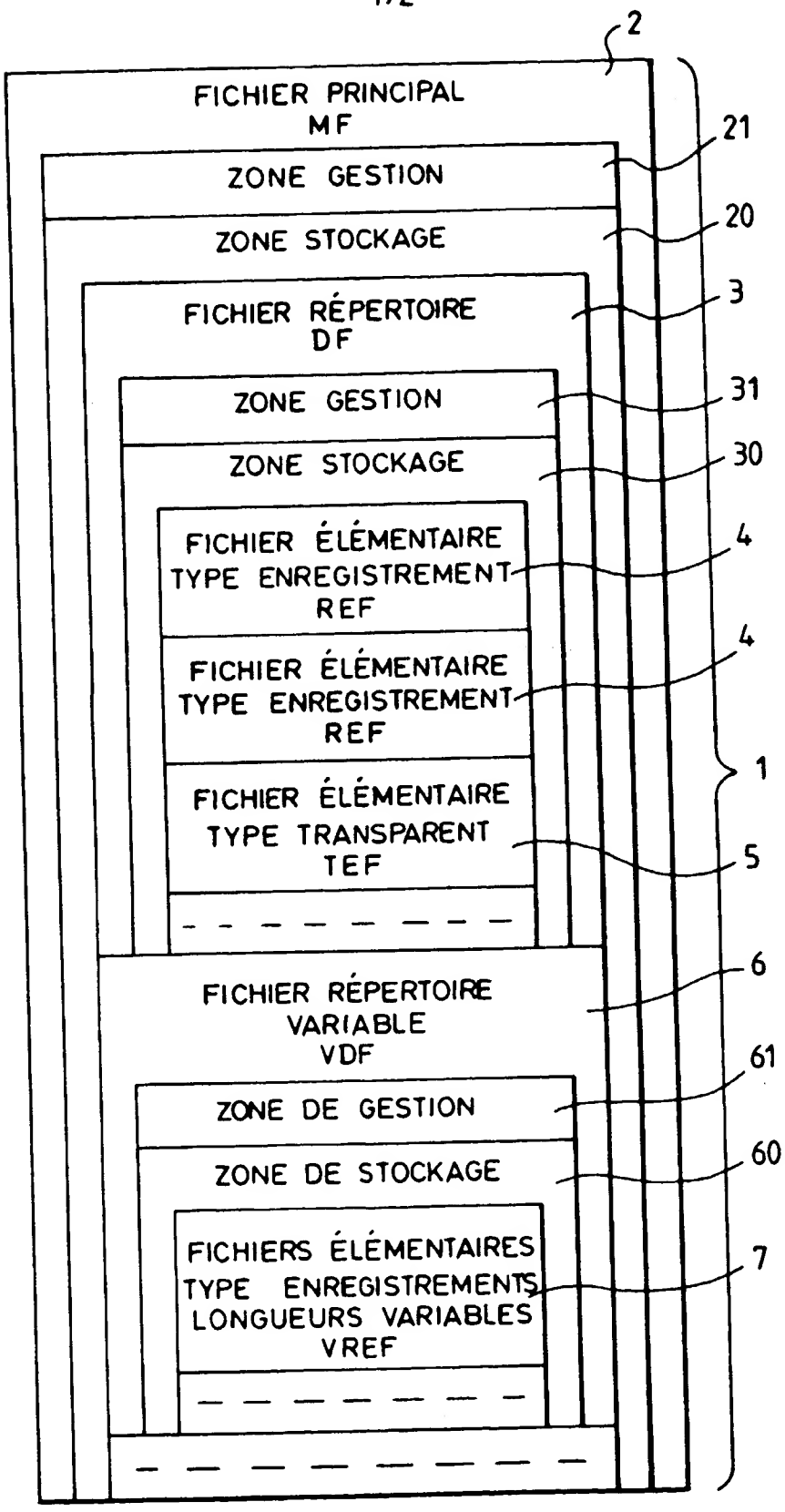


FIG.1

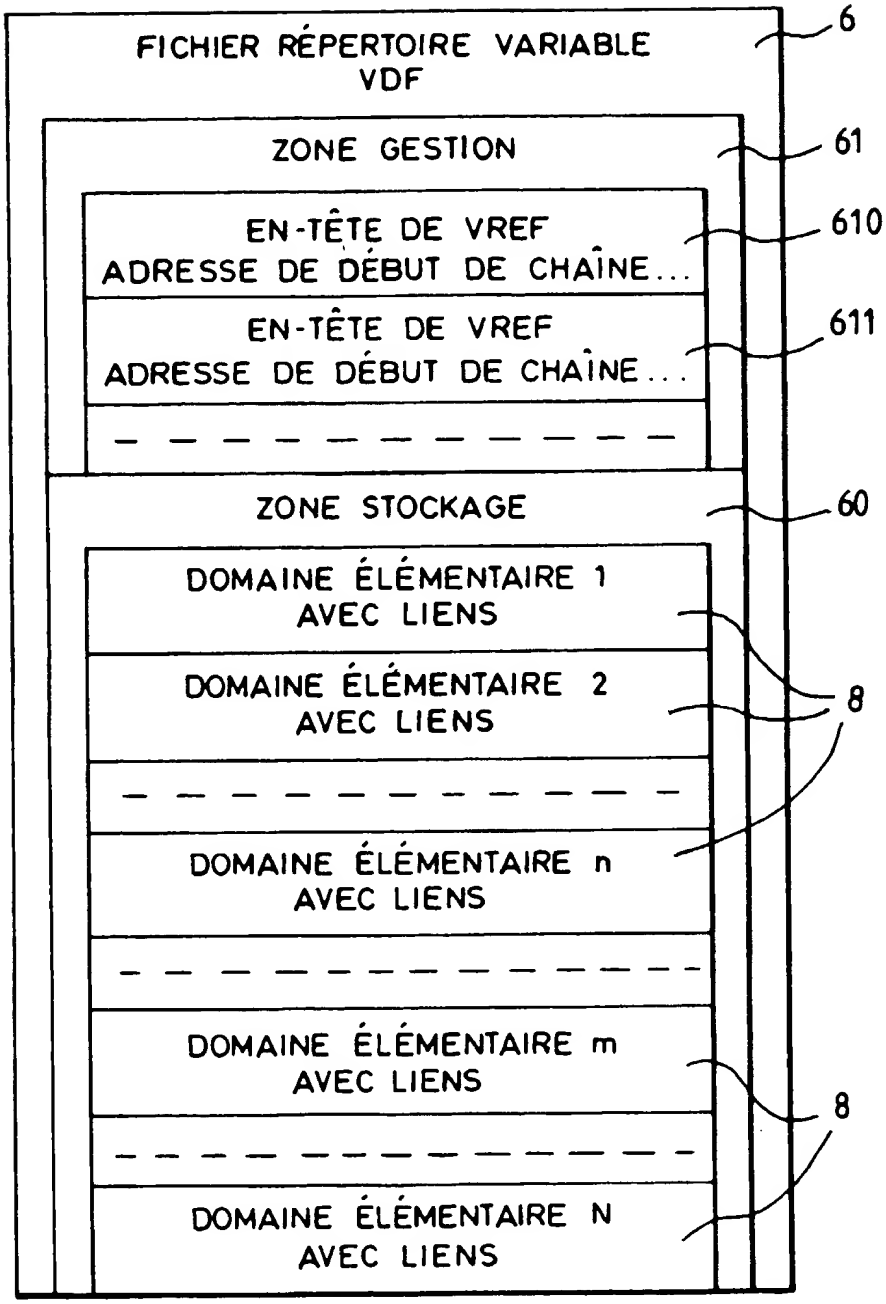


FIG.2

2759795

REPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
de la
PROPRIETE INDUSTRIELLE

RAPPORT DE RECHERCHE PRELIMINAIRE

établi sur la base des dernières revendications déposées avant le commencement de la recherche

N° d'enregistrement national
FA 540762
FR 9701735

DOCUMENTS CONSIDERES COMME PERTINENTS

Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	Revendications concernées de la demande examinée
A	EP 0 622 736 A (TOKYO SHIBAURA ELECTRIC CO) * colonne 1, ligne 1 - colonne 2, ligne 56; figures 6,11 *	1-3
A	EP 0 446 940 A (FUJITSU LTD) 18 septembre 1991 * abrégé; figure 2 *	1-3,6,8
A	US 5 365 045 A (IIJIMA YASUO) 15 novembre 1994 * colonne 4, ligne 31 - colonne 5, ligne 55 * * colonne 5, ligne 61 - colonne 6, ligne 40; figure 18A *	1,2,6 7

DOMAINES TECHNIQUES RECHERCHES (Int.CL.6)

G07F
G06F

Date d'achèvement de la recherche: 27 octobre 1997

Examinateur: Fournier, C

CATEGORIE DES DOCUMENTS CITES

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.
D : cité dans la demande
L : cité pour d'autres raisons

& : membre de la même famille, document correspondant

EPO FORM 1503 03.82 (P04C13)

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 674 290 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **94402624.4**

(51) Int. Cl.[6]: **G06K 19/073**

(22) Date of filing: **18.11.94**

(30) Priority: **25.02.94 JP 28212/94**

(43) Date of publication of application:
27.09.95 Bulletin 95/39

(84) Designated Contracting States:
DE FR GB

(71) Applicant: **FUJITSU LIMITED**
1015, Kamikodanaka,
Nakahara-ku
Kawasaki-shi,
Kanagawa 211 (JP)

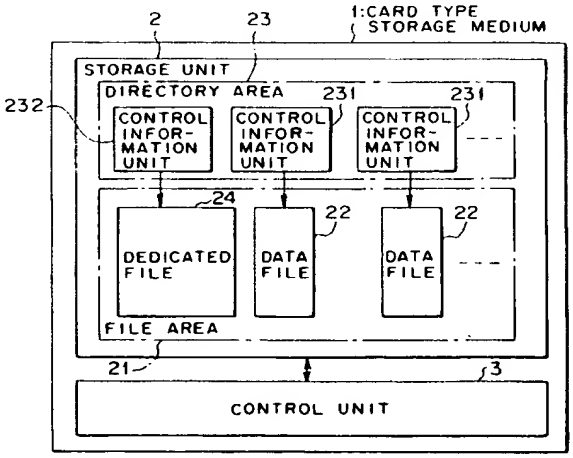
(72) Inventor: **Hoshino, Masao, c/o Fujitsu Limited**
1015, Kamikodanaka,
Nakahara-ku
Kawasaki-shi,
Kanagawa 211 (JP)

(74) Representative: **Joly, Jean-Jacques et al**
Cabinet Beau de Loménie
158, rue de l'Université
F-75340 Paris Cédex 07 (FR)

(54) Card type storage medium and card type storage medium issuing apparatus.

(57) The card type storage medium comprises a storage unit (2) holding a file area (21) including a dedicated file (24) served to hold PINs and file names of data files (22) stored in a directory area (23) in the storage unit (2) such that the PIN and file name of each data file (22) correspond to each other. The card type storage medium holds control information (232) including a master PIN for the dedicated file (24) in the directory area (23) in the storage unit (2). A recovery information unit (25) is additionally provided in a data file (22) in the file area (21) in the storage unit (2), which holds recovery information obtained every time the data file (22) is updated. This card type storage medium is applicable to, for example, an IC card.

FIG. 1



EP 0 674 290 A2

BACKGROUND OF THE INVENTION

(1) Field of the Invention

The present invention relates to a card type storage medium such as an IC card used as a cashless card, an identification (I.D.) card, a health management card, a municipal corporation card, etc. and an issuing apparatus issuing such card type storage medium.

(2) Description of the Related Art

A card type storage medium, for example, an IC card having an integrated circuit therein has been widely spread in recent years.

A file controlling program is set into the IC card to retain data therein so that the data to be processed by an external application program that is incorporated within a terminal apparatus, a host computer or the like can be managed in each file as as a unit.

Referring to FIG. 17, a typical IC card 100 comprises a terminal (a contact or a data communication mechanism) 110, a storage 120 and a control unit 130.

When the IC card 100 is inserted into an IC card reader/writer (not shown) of a terminal apparatus, a host computer or the like, the terminal 110 is brought into contact with a terminal of the IC card reader/writer to send and receive a signal.

The storage 120 has a file area in which data to be processed by each various application program is retained in each file, and a directory area 123 which retains control information about each data file 122 held in the file area 121.

The control unit (MPU: micro processor unit) 130 is to manage the data retained in the file area 121 in the storage unit 120 on the basis of the control information stored in the directory area 123 in the storage unit 120.

Some IC card has an electric source therein, and some IC card needs to be supplied an electric energy from a terminal apparatus or a host computer by being inserted into the terminal apparatus or the host computer. In the latter case, a non-volatile storage such as an EEPROM is used as the storage unit 120.

Such IC card 100 is used as a cashless card, an ID card, a health management card, a municipal corporation card, etc.

In department stores, super markets, etc., a POS system has been accomplished with employment of a cashless card such as a prepaid card or a credit card for sales promotion. If the IC card is used as such cashless card, it is essential to provide a function for advance payment or future payment to the cashless card, for example, the prepaid card or the credit card.

If the IC card 100 is used as an ID card to improve convenience in, for example, an intelligent building, the IC card needs to have a function to hold data about entrance and retrieval to and from the room, attendance of employees, etc. in the data files 122.

If the IC card 100 is used as a health management card in a hospital, a fitness facility or the like to improve convenience, the IC card 100 has to hold various data such as appointment, carte, results of examination and measurement for the management.

Likewise, if the IC card 100 is used as a municipal corporation card to improve use of public facilities or administrative service, the IC card holds data about appointment of the facilities, automatic issue of various applications as data files 122 therein.

The IC card 100 shown in FIG. 17 has predetermined personal identification number (hereinafter, referred as PIN) for every data file 122 retained in the storage unit 120 in order to reinforce the security of the data retained in the IC card 100. Each of the PIN is held as control information in the directory area 123 in the storage unit 120.

In order to gain an access from an external application or the like, only when a PIN sent with the access is in coincidence with the PIN retained in the directory area 123 in the storage unit 120, the control unit 130 allows reading or updating of the data retained in the data file 122.

The PIN for each data file 122 is set when the IC card 100 is issued by a card issuing apparatus (not shown). Management of the PIN set by the card issuing apparatus, which varies from each other depending on a card owner, is carried out by another host computer (not shown) different from the card issuing apparatus.

If a person owing the IC card accidentally forgets a PIN of his or her own IC card 100, the PIN is read out from the host computer managing the PIN through a terminal apparatus which can gain an access to the host computer to verify the PIN.

The host computer manages the PINS of the owners of all issued IC cards (card type storage media) 100. In addition, it is sometimes necessary to set plural different PINs to every data files in each IC card. The host computer therefore requires a large area in the storage to manage the PINs. The management of the PINs is, therefore, quite complex and troublesome to the entire IC card system. Moreover, in the event of an accident, use of a terminal apparatus accessible to the host computer is indispensable to verify the IC card. Such verification of the IC card causes inconvenience to

users of the IC card.

Meanwhile, a card type storage medium, which is used ahead of an IC card, for example, a magnetic card, is operated in a mode where the stored data is unchangeable as personal identification information (ID). An IC card 100 as above is used in a mode where stored data (for example, information about an amount of money) is variable, as represented by a cashless card.

In such mode of use, the IC card is used as a cashless card. In the event of an accident such as system down, power-source break-down, pull-out of the IC card 100 in the course of an updating process to receive money or for account settlement, a failure may develop in data in the data file 122 in the IC card 100. At present, it is impossible to repair or restore (data recovery) such failure in data within the IC card 100.

To cope with such event, a presently conducted technique is to set an area of 1 byte referred as BCC (block check character) in each record of data held in the data file 122 of the IC card 100, adjust the BCC such that a bit number in each record becomes an even number or an odd number and write the BCC in each record to make a check on the bit number in each record as to whether the bit number is an even number or an odd number upon reading out the data. For instance, in the case where the BCC is so adjusted that the bit number in each record is an even number, if the bit number in the record is an odd number upon checking, some action is taken upon check-out such as to prohibit the IC card 100 from being used.

It is, however, impossible to detect a system failure as conflicting data developed between the records by such BCC check, as shown in FIG. 18.

Namely, in the case where data writing and updating are executed a plurality of times (three times in FIG. 18) as one unit of process in the course from an open to close of the IC card 100 by the application program 200 of an external terminal apparatus or the like, if a system failure occurred before the second updating after the first record was updated, it is impossible to detect the system failure by the BCC since no conflict occurred in data as a record unit held in the IC card 100.

Since the BCC checks a number of bits by a record unit, if 2 bits (an even bit) are left out, or the number of bits are the same but their represented value are different (for example, "0111" and "1011", if three bits), it is impossible to detect such failure as conflicting data.

In consequence, for example, as shown in FIG. 19, if a system failure occurred while one record is being written into the IC card 100, causing a situation that there exist an updated part and an unupdated part within the same record, there is possibility that such failure cannot be detected.

To solve the above problem, it is necessary to provide a BCC in each record in the file area 121 of the storage unit 120. This results in that a large area is required for the BCCs, if a large volume of data need to be stored in the card.

As stated above, there has been developed no technique to repair and restore data failure (data recovery) within the IC card 100. If such repair and restore are handled on the side of the apparatus, the host computer needs to manage recovery information (restoration data and the like) of all IC cards every time the IC card is used. As a result, it is necessary to execute the recovery on data conflict by (1) communicating with the host computer in real time to restore the data, or (2) prohibiting the IC card from being used, and issuing a new card.

To cope with the above problem, the conventional IC card has disadvantages such that a configuration of the IC card system become difficult, a large area is required to store recovery information in the storage of the host computer, and management of the entire IC card system becomes quite complex, as same as the PIN management. Further, to repair and restore the data in the IC card 100 in the event of a system failure, it is necessary to use a terminal apparatus accessible to the host computer, or to reissue the IC card. Such data recovery work is quite troublesome to the card user.

SUMMARY OF THE INVENTION

From the above viewpoint, an object of this invention is to provide a card type storage medium and a card type storage medium issuing apparatus, in which management of PINs heretofore carried out by a host computer becomes dispensable, the PIN management in the entire system is simplified, and verification of a PIN in an event of an accident is easily and simply carried out so that inconvenience to users may be mitigated upon verification of the PIN.

Another object of this invention is to provide a card type storage medium which can detect reliably conflicting data developed due to a system failure without using a BCC, and to realize repair and restore of the conflicting data developed due to a system failure by and within the card itself, thereby simplifying the apparatus configuration and reducing inconvenience to the users upon restoring the data.

The present invention therefore provides a card type storage medium comprising a storage unit having a file area holding data in each file as a unit and a directory area holding therein control information units each including a PIN of a data file

in said file area in said storage unit on the basis of said control information units in said directory area in said storage unit, said control unit allowing an access process on a data file only when a PIN held in said control information unit in said directory area in said storage unit is in agreement with a PIN fed from outside, the improvement comprising a dedicated file being set in said file area in said storage unit, said dedicated file holding PINs of the data files held in said respective control information units in said directory area in said storage unit and file names of the data files such that the PIN and file name of each data file correspond to each other, another control information unit being set in said directory area in said storage unit, said control information unit holding a master PIN of said dedicated file.

According to the above card type storage medium of this invention, a dedicated file is provided in a file area in the storage unit to hold data including PINs and files names of the respective data files. It is, therefore, possible to manage the PINs retained in each card type storage medium by and within the card type storage medium itself. It is also possible to omit PIN management by the host computer, largely reducing a burden of the PIN management on the entire system.

As another aspect, the present invention also provides a card type storage medium issuing apparatus issuing the above card type storage medium comprising the storage unit and the control unit, said card type storage medium issuing apparatus comprising a data file creating means, in response to a data file creating command from outside, setting a control information unit for a data file including a PIN of said data file to create said data file in said file area in said storage unit according to said data file creating command, a PIN matching means, in response to a data file accessing command to gain an access to the data file created by said data file creating means from the outside, making a judgement as to whether the PIN of said data file to be accessed according to said data file access command held in said control information unit in said directory area in said storage unit is in agreement with a PIN included in said data file accessing command supplied from the outside, a data file accessing means executing an access process on the data file to be accessed when said PIN matching means judges that said two PINs are in agreement, a dedicated file creating means, in response to a dedicated file creating command from the outside, setting a control information unit for said dedicated file including a master PIN for said dedicated file to create said dedicated file in said file area in said storage unit according to said dedicated file creating command, a master PIN matching means, in response to a dedicated file access command to gain an access to said dedicated file created by said dedicated file creating means from the outside, making a judgement as to whether the master PIN of said dedicated file held in said control information unit in said directory area in said storage is in agreement with a master PIN included in said dedicated file access command supplied form the outside, and a dedicated file access means executing an access process on said dedicated file when said master PIN matching means makes a judgement that the above two master PINs are in agreement, upon issuing said IC card, said dedicated file accessing means writing the PINs of the data files held in said respective control information units in said directory area in said storage unit into said dedicated file such that the PIN and file name of each data file corresponds to each other according to a dedicated file accessing command supplied from outside after said dedicated file creating means created said dedicated file.

In the above card type storage medium issuing apparatus of this invention, upon issuing the IC card, said dedicated file creation instructing means first transfers a dedicated file creating command. Said dedicated file access instructing means then generates a dedicated file access command including data including PINs and file names of the respective data files and transfers it to said card type storage medium, thereby setting a dedicated file holding data including the PINs and file names of the respective data files such that a PIN and file name of each data file correspond to each other in the file area in the storage unit of the card type storage medium. It is, therefore, possible to manage the PINs of each card type storage medium by and within the card type storage medium itself. The management of the PINs by the host computer thus can be omitted, largely reducing a burden to manage the PIN on entire system.

The card type storage medium according to this invention comprising a storage unit having a file area holding data by file therein and a directory area holding control information about each data file in said file area therein and a control unit managing data in said area on the basis of said control information in said directory area in said storage unit, said card type storage medium executing updating on a data file by said control unit in response to an instruction from outside. A recovery information unit is additionally provided in the data file in said file area in said storage unit, into which recovery information obtained every time said control unit updates the data file is written. A start serial number obtained when a data file is opened and an end serial number obtained when the data file is closed are written into said recovery information unit as recovery information.

As still another aspect, the card type storage medium of this invention comprising a storage unit and a control unit, said control unit comprising a data file opening means opening, in response to a data file opening instruction supplied form outside, a data file in said file area in said storage unit on the basis of the control information in a directory area in said storage unit after the data file has been opened, a data file updating means updating, in response to a data file updating instruction supplied from the outside, data in a data file opened by said data file opening means, and a data file closing means closing, in response to a closing instruction supplied from the outside, the data file opened by said data file opening means after the data file has been opened. A recovery information unit is additionally provided in the data file in said file area in said storage unit, into which recovery information obtained every time said control unit updates the data file. The control unit further comprises a start serial number obtaining means obtaining a start serial number when said data file opening means opens a data file to write it into said recovery information unit, and an end serial number obtaining means obtaining an end serial number every time said data file closing means closes a data file to write it into said recovery information unit as recovery information.

According to the card type storage medium according to this invention, it is possible to detect that a system failure occurred between an open and close of a data file, by comparing the start serial number with the end serial number held in the recovery information unit.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram illustrating an aspect of this invention;

FIG. 2 is a block diagram illustrating another aspect of this invention;

FIG. 3 is a block diagram illustrating still another aspect of this invention;

FIG. 4 is a block diagram illustrating still another aspect of this invention;

FIG. 5 is a block diagram showing a card type storage medium and an issuing apparatus issuing the card type storage medium according to the first embodiment of this invention;

FIG. 6 is a block diagram showing a file structure in a storage unit of the card type storage medium according to the first embodiment;

FIG. 7 is an illustration of a typical hardware configuration of the card type storage medium issuing apparatus according to the first embodiment;

FIG. 8 is a block diagram showing a card type storage medium according to a second embodiment of this invention;

FIG. 9 is a block diagram showing a file structure in a storage unit of the card type storage medium according to the second embodiment;

FIG. 10 is an illustration showing a content of data held in a recovery information unit of the card type storage medium according to the second embodiment;

FIG. 11 is a flow chart of an operation to obtain recovery information in the card type storage medium according to the second embodiment;

FIG. 12 is a flow chart of an operation to detect a system failure and restore data in the card type storage medium according to the second embodiment;

FIGS. 13A through 13C are illustrations of a content of data in the recovery information unit in order to explain an operation of the card type storage medium according to the second embodiment;

FIG. 14 is an illustration of a content of data in the recovery information unit in order to explain the operation of the card type storage medium according to the second embodiment;

FIG. 15 is an illustration of a content of data in the recovery information unit in order to explain the operation of the card type storage medium according to the second embodiment;

FIGS. 16A and 16B are illustrations of a content of data in the application area and the recovery information unit in order to explain the operation of the card type storage medium according to the second embodiment;

FIG. 17 is a block diagram showing a configuration of a typical IC card;

FIG. 18 is an illustration showing a state of data stored in an IC card when a system failure occurred.

FIG. 19 is an illustration showing a state of data stored in an IC card when a system failure occurred.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

(1) Description of Aspects of This Invention

FIG. 1 is a block diagram illustrating an aspect of this invention. In FIG. 1, reference numeral 1 denotes a card type storage medium. The card type storage medium 1 comprises a storage unit 2 and a control unit 3.

The storage unit 2 includes a file area 21 holding data in each file as a unit and a directory area 23 having a control information unit 231 to hold a PIN for each data file 22 in the file area 21 therein.

The control unit 3 is to manage the data files 22 in the file area 21 in the storage unit 2 on the basis of data held in the control information units 231 in the directory area 23 in the storage unit 2. In this card type storage medium 1; only when the PIN held in the control information unit 231 in the directory area 23 in the storage unit 2 agrees with a PIN fed from outside, the control unit 3 permits the data files to be accessed.

In the file area 21 in the storage unit 2 of this card type storage medium 1, there is provided a dedicated file 24 to hold PINs and file names of the data files 22 such that each of the PIN corresponds to its file name of the data file 22 that is retained in the control information units 231 in the directory area 23 in the storage unit 2. Likewise, there is provided another control information unit 232 to hold a master PIN (i.e., a PIN that only the system manager knows) for the dedicated file 24 in the director area 23 in the storage unit 2.

It is possible to encipher the PINs of the data files 22 and hold them in the dedicated file 24.

In the card type storage medium shown in FIG. 1, the data of the PINs for the respective data files 22 and the corresponding file names are set in the dedicated file 24. The PINs in each card type storage medium are therefore managed by and within the card type storage medium itself. As a result, the management of the PINs by the host computer is dispensable.

The data in the dedicated file 24 cannot be read out without an input of the master PIN (that is known by only the system manager) held in the control information unit 232 in the directory area 23.

The enciphered PINs for the respective data files 22 in the dedicated file 24 can more effectively prevent the PINs for the respective data files 22 from being known by other persons except the system manager, if the master PIN gets to be known by the other person.

FIG. 2 is a block diagram illustrating another aspect of this invention. In FIG. 2, reference numeral 10 denotes a card type storage medium. The card type storage medium 10 has a similar configuration to the card type storage medium shown in FIG. 1, essentially comprising a storage unit 2 and a control unit 3.

The storage unit 2 has, as similar to that shown in FIG. 1, a file area 21 retaining data in each file as a unit and a directory area 23 including control information units 231 each retaining a PIN for a data file 22 in the file area 21 therein.

The control unit 3 is, as also similar to the one shown in FIG. 1, to manage the data files 22 in the file area 21 in the storage unit 2 on the basis of the data held in the control information units 231 in the directory area 23 in the storage unit. The control unit 3 shown in FIG. 2 is provided with a data file creating means 31, a PIN matching means 32, a data file accessing means 33, a dedicated file creating means 34, a master PIN matching means 35 and a dedicated file accessing means 36.

When receiving a data file creating command from the outside (i.e., a card type storage medium issuing apparatus 4 described later), the data file creating means 31 sets the control information unit 231 for the data file 22 containing a PIN for the data file 22 in response to the data file creating command in order to create said data file 22 in the file area 21 in the storage unit 2.

When receiving a data file access command to gain an access to the data file 22 created by the data file creating means 31 from the outside, the PIN matching means 32 makes a judgement as to whether a PIN included in the above data file accessing command agrees with the PIN of the data file 22 to be accessed held in the control information unit 231 in the directory area 23 in the storage unit 2.

The data file accessing means 33 is to gain an access to the data file 22 to be accessed when a result of the matching carried out by the PIN matching means 32 is positive.

When receiving a dedicated file creating command from the outside (i.e., the card type storage means issuing apparatus 4 described later), the dedicated file creating means 34 sets a control information unit 232 for a dedicated file 24 including a master PIN (known only by the system manager) for the dedicated file 24 in the directory area 23 in the storage unit 2 in response to the dedicated file creating command in order to create the dedicated file 24 in the file area 21 in the storage unit 2.

When receiving a dedicated file access command to gain an access to the dedicated file 24 created by the dedicated file creating means 34 from the outside (i.e., the card type storage medium issuing apparatus 4 described later), the master PIN matching means 35 makes a judgement as to whether the master PIN of the dedicated file 24 retained in the control information unit 232 in the directory area 23 in the storage unit 2 agrees with a master PIN contained in the inputted dedicated file access command.

When a result of the matching between the above two master PINs carried out by the master PIN matching means 35 is positive, the dedicated file accessing means 36 allows an access to the dedicated file 24.

Upon issuing the card type storage medium 10 of this invention, the dedicated file creating means 34, to begin with, creates the dedicated file 24. The dedicated file access means 36 next writes PINs of the respective data files 22 retained in the control

information in the directory area 23 in the storage unit 2 into the dedicated file 24 such that each of the PIN of the data file 22 corresponds to its file name, in response to the dedicated file access command supplied from the outside (the card type storage medium issuing apparatus 4, described later).

It is possible to encipher the PINs for the data files 22 and hold them in the dedicated file 24.

In FIG. 2, reference numeral 4 denotes the card type storage medium issuing apparatus. The card type storage medium issuing apparatus 4 issues the card type storage medium 10 (or a card type storage medium 1) as described hereinbefore, comprising a data file creation instructing means 41, a data file access instructing means 42, a dedicated file creation instructing means 43 and a dedicated file access instructing means 44.

The data file creation instructing means 41 sets a control information unit 231 of the data file 22 including a PIN for the data file 22 in the directory area 23 in the storage unit 2. The data file creation instructing means 41 then generates a data file creating command including the PIN, and transmit the generated data file creating command to the card type storage medium 10 (i.e., the data file creating means 31) in order to create the data file 22 in the file area 21 in the storage unit 2.

The data file access instructing means 42 generates a data file access command including a PIN for the data file 22 to be accessed, and transmits the generated data file accessing command to the card type storage medium 10 (i.e., the PIN number matching means 32 and the data file access means 32) in order to get an access to the data file 22 created in the file area 21 in the storage unit 2.

The dedicated file creation instructing means 43 sets the control information unit 232 for the dedicated file 24 including a master PIN for the dedicated file 24 in the directory area 23 in the storage unit 2. The dedicated file creation instructing means 43 generates a dedicated file creating command including the master PIN, and transmits the generated dedicated file creating command to the card type storage medium 10 (i.e., the dedicated file creating means 34) in order to create the dedicated file 24 in the file area 21 in the storage unit 2.

The dedicated file access instructing means 44 generates a dedicated file access command including the master PIN for the dedicated file 24, and transmits the generated dedicated file access command to the card type storage medium 10 (i.e., the master PIN matching means 35 and the dedicated file accessing means 36) in order to gain an access to the dedicated file 24 created in the file area 21 in the storage unit 2.

In the card type storage medium issuing apparatus 4, the dedicated file creating means 43 transfers a dedicated file creating instructing command to the card type storage medium 10 when the card type storage medium 10 is issued. Thereafter, the dedicated file access instructing means 44 generates a dedicated file access command including data of file names and PINs of the respective data files 22, and transfers it to the card type storage medium 10 (i.e., the dedicated file access means 36) in order to write the PINs of the data files 22 such that the PIN of each data file 22 corresponds to its file name that is retained in the control information unit 231 in the directory area 23 in the storage unit 2.

It is possible that, upon verification of PINs of the card type storage medium, the dedicated file access instructing means 44 of the card type storage medium issuing apparatus 4 generates a dedicated file accessing command including a master PIN, and transfers it to the card type storage medium 10 (i.e., the master PIN matching means 35 and the dedicated file accessing means 36) in order to read out data from the dedicated file 24 in the file area 21 in the storage unit 2 in the card type storage medium 10 to be verified.

When the data of the PIN and the file name of the data file 22 is read out from the dedicated file 24 in the card type storage medium 10 in response to the dedicated file accessing command from the dedicated file access instructing means 44 of the card type storage medium issuing apparatus 4, the data file access instructing means 42 generates a data file accessing command including the PIN read out, and transfers it to the card type storage medium 10 (i.e., the PIN matching means 32 and the data file access means 33) to give an instruction to the card type storage medium 10 to verify the correctness of the data file 22 corresponding to the PIN read out.

In the case where enciphered PINs of data files are held in the dedicated file 24, there are also provided an enciphering means enciphering the PINs of the data files 22 to be written into the dedicated file 24 in the card card type storage medium 10 by the dedicated file access directing means 44, and a decoding means decoding the enciphered PINs of the data files 22 read out from the dedicated file 24 in the card type storage medium 10 from the dedicated file access instructing means 44.

In the card storage medium 10 set forth above in connection with FIG. 2, the data file creating means 31 sets a control information unit 231 for the data file 22 including a PIN inthe directory area 23 in the storage unit 2 in response to a data file creating command from the card type storage mdium issuing apparatus 4.

When receiving a data file accessing command to access a data file 22 created by the data file creating means 31 from the outside, the PIN matching means 32 makes a judgement as to whether the PIN included in the data file accessing command agrees with the PIN of the data file 22 (held in the control information unit 231 in the directory area 23) to be accessed.

When a result of the matching executed by the PIN matching means 32 is positive, the data file access means 33 allows an access to the data file 22 to be accessed.

In the card type storage medium 10, the dedicated file creating means 34 sets, upon issuing the card type storage medium, the control information unit 232 for the dedicated file 24 including the master PIN (i.e., the PIN known by only the system manager) in the directory area 23 in the storage unit 2, in response to the dedicated file creating command in order to create the dedicated file 24 in the file area 21 in the storage unit 2.

When receiving the dedicated file accessing command to gain an access to the dedicated file 24 created by the dedicated file creating means 34 from the card type storage medium issuing apparatus 4, the master PIN matching means 35 makes a judgement as to whether a master PIN included in the dedicated file accessing command agrees with the mater PIN (retained in the control information unit 232 in the directory area 23) of the dedicated file 24.

When a result of the matching executed by the master PIN matching means 35 is positive, the dedicated file access means 36 carries out an access process (that is, write/read) on the dedicated file 24.

When the card type storage medium 10 is iussed, the dedicated file accessing means 36 writes a PIN and a file name of each data file into a dedicated file 24 in such a manner that the PIN and the file name correspond to each other in response to a dedicated file accessing command from the card type storage medium issuing apparatus 4 after the dedicated file creating means 34 has created the dedicated file 24.

In the above manner, data of the PIN and the file name of each data file 22 is written in the dedicated file 24 in the file area 21 in the storage unit 2 of the card type storage medium 10. The management of the PINs in each card type storage medium 10 is carried out by and within the card type storage medium 10 itself, management of the PINs by the host computer is thus dispensable.

The data in the dedicated file 24 cannot be read out without knowing the master PIN (the PIN known by only the system manager) retained in the control information unit 232 in the directory area 23.

Enciperment of the PINs of the data files 22 stored in the dedicated file 24 is more effective to prevent the PINs of the data files 22 from leaking outside as they are, even if the master PIN gets to be known by another person except the system manager.

The above mentioned card type storage medium issuing apparatus 4 shown in FIG. 2 issues the card type storage medium 10 (or a card type storage medium 1).

More specifically, the data file creation instructing means 41 generates a data file creating command including PINs of data files 22, and transfers it to the card type storage medium 10 (i.e., the data file creating means 31) to set the control information unit 231 for the data files 22 including the PINs for the respective data files 22 in the directory area 23 in the storage unit 2, in response to the data file creating command so that the data files 22 may be created in the file area 21 in the storage unit 2.

The data file access instructing means 42 generates a data file accessing command including a PIN for a data file 22 to be accessed, and transfers it to the card type storage medium 10 (i.e., the PIN matching means 32 and the data file accessing means 32) to perform an access process (i.e., write/read) on the data file 22 created in the file area 21 in the storage unit 2.

On the other hand, the dedicated file creation instructing means 43 generates a dedicated file creating command including a master PIN, and transfers it to the card type storage medium 10 (i.e., the dedicated file creating means 34) to set the control information unit 232 for the dedicated file 24 including the master PIN for the dedicated file 24 so that the dedicated file 24 is created in the file area 21 in the storage unit 2.

The dedicated file access instructing means 44 generates a dedicated file accessing command including the master PIN for the dedicated file 24, and transfers it to the card type storage medium 10 (i.e., the master PIN matching means 35 and the dedicated file access means 36) to perform an access process (i.e., write/read) on the dedicated file 24.

Upon issuing the card type storage medium 10 (or the card type storage medium 1), the dedicated file creation instructing means 43, to begin with, transfers the dedicated file creating command. The dedicated file access instructing means 44 next generates a dedicated file accessing command including data of the PINs and the file names of the respective data files 22, and transfer it to the card type storage medium 10 (i.e., the dedicated file accessing means 36).

The data of the PINs and the file names of the respective data file 22 is set in the dedicated file 24 in the file area 21 in the storage unit 2 of the

card type storage medium 10. The management of the PINs in the card type storage medium 10 is carried out by and within each card type storage medium 10 itself, the management of the PINs by the host computer may thus be omitted.

The data in the dedicated file 24 cannot be read out by a person not knowing the master PIN (i.e., the PIN known only by the system manager) retained in the control information unit 232 in the directory area 23.

In the event of an accident, the dedicated file access instructing means 44 generates a dedicated file access command including the master PIN, and transfers it to the card type storage medium 10 (i.e., the master PIN matching means 35 and the dedicated file accessing means 36), whereby the card type storage medium issuing apparatus 4 can read out the data (i.e., the data of the PINs and the file names of the respective data files 22) from the dedicated file 24 in the file area 21 in the storage unit 2 of the card type storage medium 10 to verify the PINs of the card type storage medium 10.

When the data of the PINs and the file names of the data files 22 is read out from the dedicated file 24 in response to the dedicated file accessing command from the dedicated file access instructing means 44, the data file access instructing means 42 generates a data file accessing command including the PIN read out, and transfers it to the card type storage medium 10 (i.e., the PIN matching means 32 and the data file accessing means 33) to verify the correctness of the data file 22 corresponding to the PIN.

It is possible to encipher PINs of the data files to be written in the dedicated file 24 by the dedicated file access instructing means 44, and decipher the enciphered PINs read out from the dedicated file 24 by the dedicated file access instructing means 44. The dedicated file can therefore hold the enciphered PINs for the respective data files 22. If the master PIN is known by another person except the system manager, the enciphered PINs of the data files are securely prevented from being known as they are.

According to this invention, since the PINs of the data files 22 and their file names are held in the dedicated file 24 in the file area 21 in the storage unit 2 of the card type storage medium 1 or 10 in such a manner that the PIN and the file name of each data file 22 correspond to each other, as stated above. Therefore, each card type storage medium 1 or 10 can manage the PINs by and within the card type storage medium itself, the management of the PINs by the host computer becomes thus dispensable and the burden to manage the PINs in the entire system can be largely reduced.

The data in the dedicated file 24 is exhibited to be read out without use of the mater PIN known by only the system manager. Moreover, the enciphered PINs of the respective data files 22 held in the dedicated file 24 can be effectively prevented from being known as they are by the other person, even if the master PIN gets to be known by the other person except the system manager. In which case, it is impossible to decipher the enciphered PINs as long as the manner of the enciperment is in secret. This can surely prevent the PINs from leaking outside, causing no trouble in security, even if the card type storage medium 1 or 10 manages the PINs therein.

Also according to this invention, when the card type storage medium issuing apparatus 4 issues the card type storage medium 1 or 10, the dedicated file creation instructing means 43 transfers a dedicated file creating command, the dedicated file access instructing means 44 then generates a dedicated file accessing command including data of the PINs and file names of the respective data files 22 to transfer it to the card type storage medium 1 or 10, whereby a dedicated file 24 holding the PINs and the file names of the respective data files 22 therein can be set so that the card type storage medium 1 or 10 can manage the PIN by itself. This can omit the management of the PINs by the host computer, largely simplifying the PIN management in the entire system. The data in the dedicated file 24 cannot be read out without the master PIN known by only the system manager.

In order to read out data of the PINs and the file names of the data files 22 from the dedicated file 24 of the card type storage medium 1 or 10, the dedicated file access instructing means 44 generates a dedicated file accessing command including the master PIN, and transfers it to the card type storage medium 1 or 10. In the event of an accident, it is possible to verify the PINs in the card type storage medium 1 or 10, mitigating inconvenience to the user upon verification of the PINs.

On the verification of a PIN, the data file access instructing means 44 transfers a data file accessing command to the card type storage medium in order to verify the PIN read out from the dedicated file 24. This process make it possible to verify the correctness of the data file 22 corresponding to the PIN read out, with a high reliability in the PIN verification process.

The PINs to be written into the dedicated file 24 by the dedicated file access instructing means 44 are enciphered by the enciphering means, while the enciphered PINs read out from the dedicated file 24 by the dedicated file access instructing means 44 are enciphered by the enciphering means, whereby the dedicated file 24 can hold the PINs for the respective data file 22 as ciphers. If

the master PIN gets to be known by person except the system manager, it is possible to prevent the PINs of the data files from leaking out as they are. So long as the manner of the encipherment does not leak out, it is impossible to decipher the PINs. This can prevent, with certain, the PINs from leaking outside, causing no trouble in security, even if the card type storage medium 1 or 10 manages the PIN therein.

FIG. 3 is a block diagram showing another aspect of this invention. In FIG. 3, reference numeral 11 denotes a card type storage medium, comprising a storage unit 2 and a control unit 5.

The storage unit 2 has a file area 21 holding data by files therein and a directory area 23 holding control information about each data file 22 in the file area 21 therein. The control unit 5 manages data in the file area 21 in the storage unit 2 on the basis of the control information in the directory area 23 in the storage unit 2. In the card type storage medium 11, the control unit 5 updates the objective data file 22 when receiving a command from outside.

The card type storage medium 11 is additionally provided a recovery information unit 25 in the data file 22 in the file area 21 in the storage unit 2, into which recovery information obtained every time an updating operation is performed on the objective data file 22 by the control unit 5 is written. In the recovery information unit 25, there are written (1) a start serial number obtained when the objective data file 22 is opened, (2) restoration data consisting of a record number to be updated and unupdated data at the record number obtained when the data file is updated, and (3) an end serial number obtained when the data file 22 is closed.

Meanwhile, it is possible to attach check serial numbers as recovery information, before and after the start serial number, the restoration data and the end serial numbers, respectively, in the recovery information unit 25.

If the updating process is performed a plurality of times on the same record number in the course from an open to close of the objective data file 22, the restoration data is not written into the recovery information unit 25 after the second updating process and later.

It is also possible to set information about a presence of the recovery information unit 25 in the objective data file 22 and information about a relative position of the recovery information unit 25 in the objective data file 22, if the recovery information unit 25 exists, in the control information unit in the directory area 23 in the storage unit 2.

In the card type storage medium shown in FIG. 3, (1) a start serial number obtained when an objective data file 22 is opened, (2) restoration data consisting of a record number to be updated and unupdated data at the record number obtained when the objective data file 22 is updated, and (3) an end serial number obtained when the objective data file 22 is closed, are written as recovery information into the recovery information unit 25 additionally provided in the data file 22 in the file area 21 in the storage unit 2.

The start serial number in the recovery information unit 25 is compared with the end serial number. If a result of the comparison is in disagreement, it is possible to know from the result an occurrence of a system failure between an open and close of the objective data file 22, without using a BCC.

Moreover, the check serial numbers are attached before and after the start serial number, the restoration data and the end serial numbers, respectively, in the recovery information unit 25 as recovery information. The check serial numbers attached before and after the start serial number, the recovered number and the end serial number are compared with each other, respectively. If a result of the comparison is in disagreement, it is also possible to detect an occurrence of system failure in the course of writing the recovered data or the end serial number into the recovery information unit 25 so as to know the effectivity of each data stored in the recovery information unit 25.

If the same record number is updated plural times between an open and close of the objective data file 22, the restoration data obtained is not written into the recovery information unit 25 after the second updating process and later. It is therefore possible to always hold a preceding data (data before the updating) before the open of the data file 22 as restoration data at the same record number in the recovery information unit 25.

The above process enables the state inside the card type storage medium 11 after an occurrence of system failure to be effectively recovered to the state before the updating process where the system failure occurred.

By setting information as to the presence of the recovery information unit 25 in a data file and information about a relative position of the recovery information unit 25 in the objective data file 22, if the recovery information unit 25 exists, in the directory area 23 in the storage unit 2, it is possible to make a judgement as to whether a predetermined data should be written into the recovery information unit 25 or data recovery should be executed on the basis of the data stored in the recovery information unit 25.

FIG. 4 is a block diagram showing still another aspect of this invention. In FIG. 4, reference numeral 12 denotes a card type storage medium corresponding to the second invention. The card type storage medium 12, as similar to the card

type storage medium 11 shown in FIG. 3, essentially comprises a storage unit 2 and a control unit 5.

The storage unit 2, as similar to the storage unit 2 shown in FIG. 3, has a file area 21 holding data in each file as a unit, and a directory area 23 holding control information including PINs for data files 22 in the file area 21 therein. In the data file 22 in the file area 21 in the storage unit 2 of this invention, there is additionally provided a recovery information unit 25 holding recovery information therein obtained every time the control unit 5 executes an updating operation on the data file 22.

The control unit 5 manages the data files 22 in the files area 21 in the storage unit on the basis of the control information in the directory area 23 in the storage unit 2, similarly to the one shown in FIG. 3. The control unit 5 of this invention includes a data file opening means 51, a data file updating means 52, a data file closing means 53, a start serial number obtaining means 54, a restoration data obtaining means 55 and an end serial number obtaining means 56.

When receiving an opening command from the outside, the data file opening means 51 opens a data file in the file area 21 in the storage unit 2 on the basis of the control information stored in the directory area 23 in the storage unit 2, in response to the opening command.

When receiving an updating command from the outside after the data file 22 has been opened, the data file updating means 52 updates data in the data file 22 that has been opened by the data file opening means 51.

When receiving a closing command from the outside after the data file 22 has been opened, the data file closing means 53 closes the data file 22 that has been opened by the data file opening means 51.

The start serial number obtaining means 54 obtains a start serial number when the data file 22 is opened by the data file opening means 51, and writes it as recovery information into the recovery information unit 25.

When the data file 22 is updated by the data file updating means 52, the restoration data obtaining means 55 obtains restoration data that is data before execution of the updating operation by the data file updating means 52, and writes it as recovery information into the recovery information unit 25.

The end serial number obtaining means 56 obtains an end serial number when the data file closing means 53 closes the data file 22, and writes it as recovery information into the recovery information unit 25.

Incidentally, it is possible to attach check serial numbers as recovery information before and after a start serial number, recovery data and end serial number, respectively, when the start serial number obtaining means 54, the recovery data obtaining means 55 and the end serial number obtaining means 56 obtain the start serial number, the restoration data and the end serial number, respectively, then write them together into the recovery information unit 25.

It is also possible to provide in the control unit 5 a restoration data initializing means initializing the restoration data obtained in the last process stored in the recovery information unit 25 before writing the new restoration data obtained this time by the recovery data obtaining means 55 into the recovery information unit 25.

When the data file updating means 52 carries out updating process a plurality of times on the same record number between an open and close of the objective data file 22, the recovered data obtaining means 55 does not write the restoration data into the recovery information unit 25 after the second updating and later.

It is also possible to set information about the presence of the recovery information unit 25 in data file 22 and information about a relative position of the recovery information unit 25 in the data file 22, if the recovery information unit 25 exists, in the control information unit for the data file 22 in the directory area 23 in the storage unit 2.

It is also possible to provide a failure detecting means in the control unit 5, which detects a failure in the last process, on the basis of a start serial number, an end serial number and check serial numbers attached before and after the start serial number and the end serial number, respectively, as recovery information stored in the recovery information unit 25, by referring to the control information about an objective data file in the directory area 23 in the storage unit 2 if the data file 22 has the recovery information unit 25, in response to an opening command from the outside.

If the check serial numbers attached before and after the start serial number are in disagreement, the failure detecting means judges that a failure occurred when the data file was opened by the data file opening means 51 in the last process. When detecting a failure that occured when the data file was opened in the last process, the failure detecting means outputs a demand to perform once more the last process and a demand to restore the start serial number.

It is possible to provide a start serial number restoring means in the control unit 5, which restores the start serial number stored in the recovery information unit 25 to the one at the time of two updating processes earlier in response to the demand to restore the start serial number from the failure detecting means.

In the case where the check serial numbers attached before and after the start serial number are in agreement but the check serial numbers attached before and after the end serial numbers are in disagreement, the failure detecting means judges that a failure occurred when the data file was closed by the data file updating means 52 in the last process. When detecting that a failure occured when the data file was closed in the last process, the failure detecting means outputs a demand to restore the end serial number.

It is possible to provide an end serial number restoring means in the control unit 5, which restores the end serial number stored in the recovery information unit 25 to the one at the time of the last process in response to a demand to restore the end serial number from the failure detecting means.

In the case where the check serial number attached before and after the start serial number and the check serial numbers attached before and after the end serial numbers are individually in agreement but the start serial number and the end serial number are in disagreement, the failure detecting means judges that a failure occurred in the course of updating the data file by the data file updating means 52 in the last process. When detecting that a failure occured in the course of updating the data file in the last process, the failure detecting means outputs a demand to once more perform the last process and a demand to restore the data in the data file 22.

It is possible to provide a data restoring means in the control unit 5, which restores the data in the data file 22 on the basis of the restoration data stored in the recovery information unit 25 in response to a demand to restore the data in the data file 22 from the failure detecting means.

The data restoring means comprises a restoration data effectiveness detecting means which makes a judgement that the restoration data is effective when the check serial numbers attached before and after the restoration data are in agreement, and a restoration data writing means which, when the recovered data effectiveness detecting means judges that the restoration data is effective, writes said restoration data before the updating as data at the record number of said restoration data in the data file 22 into the data file 22.

In the card type storage medium 12 shown in FIG. 4, recovery information obtained every time the data file 22 is updated according to an instruction form the control unit 5 is written in the recovery information unit 25, which is provided in the data file 22 in the file area 21 in the storage unit 2.

More specifically, a start serial number obtained by the start serial number obtaining means 54 when the data file opening means 51 opens the data file 22, restoration data consisting of a record number to be updated and unupdated data at the same record number obtained by the restoration data obtaining means 55 when the data file updating means 52 updates the data file 22, and an end serial number obtained by the end serial number obtained means 56 when the data file closing means 53 closes the data file 22 are written in the recovery information unit 25.

The start serial number and the end serial number in the recovery information unit 25 are compared with each other. If the two numbers are in disagreement, it means that a system failure occurred between an open and close of the data file 22. It is thus possible to detect a system failure without using a BCC.

When the start serial number obtaining means 54, the restoration data obtaining means 55 and the end serial number obtaining means 56 obtain a start serial number, restration data and an end serial number, respectively, check serial numbers are attached before and after the start serial number, the restoration data and the end serial number, respectively, then written into the recovery information unit 25 as recovery information. If the check serial numbers of the start serial number, the recovered data and the end serial number are in disagreement when compared with each other, it means that a system failure occurred while the start serial number, the recovered data or the end serial number are written into the recovery information unit 25 so that it becomes possible to detect a system failure and to verify the effectiveness of data stored in the recover information unit 25.

Before the restoration data obtained by the restoration data obtaining means 56 is written into the recovery information unit 25, the restoration data obtained in the last process stored in the recovery information unit 25 is initialized by the recovery data initializing means, thereby preventing the previously stored restoration data from remaining in the recovery inforamtion unit by overwriting the new restoration data obtained in this process when the recovered data is written into the recovery information unit 25, further preventing an erroneous detection of a system failure or the like.

In the case where the updating process is carried out a plurality of times by the data file updating means 52 on the same record number between an open and close of the data file 22, the recovered data obtaining means 55 does not write the recovered data into the recovery information unit 25 after the second updating process and later. Whereby, the previous data obtained before the open of the data file 22 (data before the updating process) of the same record number may be always held as restoration data in the recovery information unit 25.

It is possible to restore the state of the card type storage medium 12 after a system failure has occured to the state before an updating process in which the system failure occurred, on the basis of the data in the recovery information unit 25.

By setting information about the presence of the recovery information unit 25 and information about a relative position of the recovery information unit 25 in a data file 22, if the recovery information unit 25 exists, in the directory area 23 in the storage unit 2, it is possible to make a judgement as to whether predetermined data should be written into the recovery information unit 25 or data recovery on the basis of the data in the recovery information unit 25 should be performed, only by referring to the directory area 23 in the storage unit from the control unit 5.

In response to an opening command from the outside, the control information stored in the directory area 23 in the storage unit 2 with respect to the objective data file 22 is referred to. If the objective data file 22 has a recovery information unit 25, the failure detecting means provided in the control unit 5 checks as to whether a failure occurred in the last process or not, on the basis of a start serial number, an end serial number and check serial numbers attached before and after the start serial number and the end serial number, thereby detecting conflicting data having developed due to a system failure without using a BCC.

If the check serial numbers attached before and after the start serial number are in disagreement, the failure detecting means detects an occurrence of a failure when the data file was opened by the data file opening means 51 in the last process. When detecting a failure, the failure detecting means outputs a demand to execute once more the last process and a demand to restore the start serial number, thereby appropriately performing again the last process that was erroneously terminated due to the system failure.

The start serial number recoverying means restores the start serial number stored in the recovery information unit 25 to the one at the time of the last process so that the state of the recovery information unit 25 can be automatically restored to the state at the time of two updating process earlier within the card type storage medium 12.

In the case where the check serial numbers attached before and after the start serial number are in agreement but the check serial numbers attached before and after the end serial number are in disagreement, the failure detecting means judges that a failure occurred when the data file closing means closed the data file in the last process. If detecting a failure, the failure detecting means outputs a demand to restore the end serial numbers to cause the end serial number restoring means to restore the end serial number stored in the recovery information unit 25 to the one at the time of the last process, thereby automatically restore the state of the recovery information unit 25 to the state at the time of the last process within the card type storage medium 12.

In the case where the check serial numbers attached before and after the start serial number and the check serial numbers attached before and after the end serial number are individually in agreement but the start serial number and the end serial number are in disagreement, the failure detecting means judges that a failure occurred during the last updating process carried out by the data file updating means 52. The failure detecting means outputs a demand to perform once more the last process and a demand to restore the data in the data file 22, thereby once again carrying out the last process that resulted in unsuccess due to the system failure.

The data restoring means restores the data in the data file 22 on the basis of the recovered data stored in the recovery information unit 25. This enables the data file 22 to be automatically restored to the state at the time of the two updating processes earlier (a state before the failure occured), thereby once more executing the last process appropriately on the data file 22 in the state two updating processes earlier.

When the data restoring means restores the data, the restoration data writing means writes only effective restoration data whose check serial numbers attached before and after the restoration data are judged by the restoration data effectiveness detecting means, thereby enabling data recovery without using restoration data in which a system failure occured during writing it (that is, data whose check serial numbers attached before and after the data are in disagreement).

According to this invention, the start serial number and the end serial number in the recovery information unit 25 are compared with each other in the card type storage medium 11 or 12. If the start serial number and the end serial number are not in disagreement, it is thus possible to detect a system failure that occurred between an open and close of the data file 22. The check serial numbers attached before and after each data are also compared with each other. If the check serial numbers are in disagreement, it is thus possible to detect a system failure that occurred while a start serial number, recovered data or an end serial number are written into the recovery information unit 25, whereby the effectiveness of each data written in the recovery information unit 25 can be verified, further conflicting data developed due to the system failure can be surely detected.

Before the recovered data obtained by the recovery data obtaining means 55 is written in the recovery information unit 25, recovered data that was written in the recovery information unit 25 in the last process is initialized by the restoration data initializing means, thereby preventing the previous restoration data from remaining in the recovery information unit 25 by overwriting when the new restoration data is written in the recovery information unit 25, further preventing the system failure from being erroneously detected.

In the case where updating process is executed a plurality of times on the same record number by the data file updating means 52 between an open and close of the data file 22, the restoration data is not written in the recovery information unit 25 after the second updating and later. The recovery information unit 25 therefore can always hold data before the data file is opened (that is, data before the updating) at the same record number. It is thus possible, even after a system failure has occcured, to effectively restore the state in the card type storage medium 1 or 12 to a state before the updating process in which a system failure occurred, on the basis of the data in the recovery information unit 25.

Information about the presence of the recovery information unit 25 and information about a relative position of the recovery information unit 25 in the data file 22, if the recovery information unit 25 exists, are set in the directory area 23 in the storage unit 2, whereby a judgement can be made on as to whether predetermined data should be written into the recovery information unit 25, or data recovery should be performed on the basis of the data in the recovery information unit 25, only by referring to the directory area 23 in the storage unit 2 from the control unit 5.

The failure detecting means can detect a failure having occurred in the last process on the basis of a start serial number, an end serial number and check serial numbers attached before and after the start serial number and the end serial number, respectively, held in the recovery information unit 25 in response to an opening command from the outside, if the data file 22 has the recovery information unit 25, thereby automatically detecting conflicting data developed due to a system failure without using a BCC within the card type storage medium 11 or 12.

According to a result of detection carried out by the failure detecting means, the start serial number restoring means, the end serial number restoring means, the data restoring means can automatically repair and restore the recovery information unit 25 or the application area, thereby simplifying the configuration of the system, reducing inconvenience to the users upon restoring the data, in addition.

When the data restoring means restores the data, the restoration data writing means writes only effective restoration data, whose check serial numbers attached before and after the restoration data are judged to be in agreement by the restoration data effective detecting means, into the data file. This makes it possible to store only certain and effective data, avoiding use of restoration data in which a system failure occurred while the restoration data was being written.

(b) Description of First Embodiment

Description will be hereinafter made in detail of a first embodiment of this invention. Now, referring to FIG. 7, there is shown an IC (integrated circuit) card 6 as a card type storage medium, having an IC unit 60 therein. The IC card 6 is issued by a card issuing apparatus (a card type storage medium issuing apparatus) 7 having a hardware configuration as shown in FIG. 7.

The IC card issuing apparatus 7 comprises, as shown in FIG. 7, an IC card reader/writer 71, a personal computer 72 and a printer 73.

The IC card reader/writer 71 is connected to the personal computer 72 via a dedicated line (for example, RS232C cable), into which the IC card 6 is inserted to be read out or written into.

The personal computer 72 causes the IC card reader/writer 71 to write data into the IC card 6 so that the IC card may have a predetermined data content therein, functioning as a main frame of the card issuing apparatus.

The printer 73 is served to printout a PIN and the like that is a result of verification made on personal identification numbers (PINs) onto a predetermined printing paper 75 in response to an instruction from the personal computer 72.

An internal configuration of the IC card 6 as a card type storage medium according to the first embodiment of this invention will be hereinafter described referring to FIGS. 5 and 6.

As shown in FIG. 5, the IC card 6 according to the first embodiment comprises a data communication mechanism 61, a storage unit 62 and a control unit 63.

The data communication mechanism 61 of the IC card 6 sends and receives information to and from the card issuing apparatus 7 when the IC card 6 is inserted into the IC card reader/writer 72 (or another terminal apparatus, a host computer, etc.) of the card issuing apparatus 7, including a terminal (a contact) which contacts with a terminal (of a data communication mechanism 710) of the IC card reader/writer 71 to transmit and receive signals thereto and therefrom.

The storage unit 62, employing a tree structure, includes a file area 621 including data to be processed by various application programs (i.e., programs to be incorporated in a terminal apparatus, a host computer, etc.) by files therein, and a directory area 623 holding control information about each file in the file area 621 therein.

For instance, data files (in an application using area) 622-1, 622-2, ..., to be handled by various application programs are held in the file area 621 in the storage unit 62, as shown in FIG. 6.

In this embodiment, the data file 622-1 is to be processed by an application program for cashless service, dedicated to hold data about, for example, the balance, use record, etc. therein. The data file 622-2 is to be processed by an application program for medical examination service in hospital or the like, dedicated to hold, for example, examination record, blood type, etc. therein.

The directory area 623 contains control information units 623-1, 623-2, ... about the respective data files 622-1, 622-2, ... therein.

In each of the control information units 623-1, 623-2, ..., a file name (APL-1 or APL-2), a position (a point or an address) in the file area 621, and a PIN (PIN; ABCD, or EFGH) of each of the data files 622-1, 622-2, ... are written.

The IC card 60 also has a dedicated file (different from the data files 622-1, 622-2, ..., for the application programs) in the file area 621 in the storage unit 62, used to manage PINs of the data files 622-1, 622-2, ..., retained therein, as shown in FIG. 6.

The dedicated file 624 holds data of file names (APL-1, APL-2, ...) of the data files (622-1, 622-2, ...) and their PINs (ABCD, EFGH, ...) in an enciphered form (PIN:****, ####, ...) in such manner that the each enciphered PIN of a data file corresponds to its file name.

The directory area 623 in the storage unit 62 has a control information unit 623-0 for the dedicated file 24, in which a position (a point or an address) in the file area 621, a file name (Master), and a master personal identification number (a PIN known by only the system manager, hereinafter referred, occasionally, as a master PIN) of the dedicated file 624 are held.

As shown in FIG. 5, the control unit (MPU: micro processor unit) 63 of the IC card 6 is to manage data retained in the file area 621 in the storage unit 62 according to the control information held in the directory area 623 in the storage unit 62. The control unit 63 according to this embodiment, comprises a data file creating unit 631, a PIN matching unit 632, a data file accessing unit 633, a dedicated file creating unit 634, a master PIN matching unit 635 and a dedicated file accessing unit 636.

The data file creating unit 631 sets the control information units 623-1, ... including pins, points and files names of the respective data files 622-1, ... in the directory area 623 in the storage unit 62 in response to a data file creating command from the card issuing apparatus 7 to create the data files 622-1, ... in the file area 621 in the storage unit 62.

In response to a data file access command from outside (i.e., the card issuing apparatus 7, the terminal apparatus, the host computer, etc.) to access a data file 622-1 created by the data file creating unit 631, the PIN matching unit 632 makes a judgement as to whether a PIN of the data file 622-1, ... that is an object of the access command (i.e., a PIN held in the control information unit 623-1 in the directory area 623) is in agreement with a PIN fed from the outside included in the above data file access command.

The PIN matching unit 632 according to this embodiment also has a file locking function to lock a data file 622-1, ... that is an object of an access when the matching of the PINs is successively resulted in failure predetermined times, and a function to release the file locking state in response to an instruction from the outside, in addition to the above matching function.

The data file access unit 633 executes an access process (write/read) on the data file 622-1 that is an object of the access when the PIN matching unit 632 judges that the PINs are in agreement.

The dedicated file creating unit 634 sets a control information unit 623-0 including a master PIN, a point and a file name of a dedicated file 624 in the directory area 623 in the storage unit 62 in response to a dedicated file creating command from the card issuing apparatus 7 to create the dedicated file 624 in the file area 621 in the storage unit 62.

In response to a dedicated file access command from the card issuing apparatus 7 to gain an access to the dedicated file 624 created by the dedicated file creating unit 634, the mater PIN matching unit 635 makes a judgement as to whether a master PIN held in the control information unit 623-0 in the directory area 623 is in agreement with a master PIN included in the above dedicated file access command from the card issuing apparatus 7.

When the master PIN matching unit 635 judges that the two PINs are in agreement, the dedicated file access unit 636 executes an access process (write/read) on the dedicated file 624.

At the time of issuing the IC card 6, the dedicated file creating unit 634, to begin with, creates the dedicated file 624, the dedicated file access unit 636 next writes PINs (enciphered PINs in this embodiment) and file names of the respective data

files 622-1, ... into the dedicated file 624 such that each PIN of the data file corresponds to its file name, in response to a dedicated file access command supplied from the card issuing apparatus 7.

The IC card may have an electric source therein, or may be supplied electric energy from the card issuing apparatus 7, the terminal apparatus or the host computer when the IC card is inserted into them. In the latter case, a non-volatile storage such as an EEPROM is employed to the storage unit 62 of the IC card 6.

A configuration of the card issuing apparatus 7 which issues a card type storage medium according to the first embodiment will be next described in detail, referring to FIG. 5.

As stated above in connection with FIG. 7, the card issuing apparatus 7 according to the first embodiment comprises the IC card reader/writer 71, the personal computer 72 and the printer 73.

The IC card reader/writer 71 has the data communication mechanism 710 which is adapted to communicate between the IC card 6 and the personal computer 72 to write or read information to and from the IC card 6 when the IC card 6 is inserted thereinto. The data communication mechanism 710 includes a terminal (a contact) contacting with a terminal (of a data communication mechanism 61) of the IC card 6 to send or receive signals.

The personal computer 72 functions as a card issuing apparatus to issue the above IC card 6 as mentioned above, comprising a data file creation instructing unit 721, a data file access instructing unit 722, a dedicated file creation instructing unit 723, a dedicated file access instructing unit 724, an enciphering unit 725 and a decoding unit 726.

The data file creation instructing unit 721 sets control information units 623-1 including PINs, points, file names of respective data files 622-1, ... in the directory area 623 in the storage unit 62. In order to create the data files 622-1, ... in the file area 621 in the storage unit 62, the data file creation instructing unit 721 generates a data file creating command including control information such as the PINs of the data files on the basis of data file creating data (including the PINs) stored in the floppy disk 76, and transfers the generated data file creating command to the data file creating unit 631 of the IC card 6 via the data communication mechanisms 710 and 61.

The data file access instructing unit 722 generates a data file access command including a PIN for a data file 622-1 to be accessed in order to execute an access process on the created data file 622-1 in the file area 621 in the storage unit 62 of the IC card 6, then transfers the generated data file access command to the IC card 6 (the PIN matching unit 632 and the data file access unit 632). The data file access instructing unit 722 also has a function to transfer an instruction to verify the correctness of the data file corresponding to the PIN when the PIN in the IC card 6 is checked, as described later.

The dedicated file creation instructing unit 723 sets a control information unit 623-0 including a master PIN, a point, a file name of the dedicated file 624 in the directory area 623 in the storage unit 62 of the IC card 6. In order to create the dedicated file 624 in the file area 621 in the storage unit 62 of the IC card 6, the dedicated file creation instructing unit 623 generates a dedicated file creating command including the master PIN on the basis of dedicated file creating data stored in the floppy disk 77, then transfers the generated dedicated file creating command to the dedicated file creating unit 634 of the IC card 6 via the data communication mechanisms 710 and 61.

In order to execute an access process on the dedicated file 624 created in the file area 621 in the storage unit 62 of the IC card 6, the dedicated file access instructing unit 724 generates a dedicated file access command including the master pin of the dedicated file 624, then transfers the generated dedicated file access command to both the master PIN matching unit 635 and the dedicated file access unit 636 of the IC card 6 via the data communication mechanisms 710 and 61. The dedicated file access instructing unit 624 also has a function to transfer an instruction to verify the PINs of the IC card 6, as described later.

The dedicated file access instructing unit 724 according to this embodiment also has a function to generate a dedicated file access command including data, a pair of the enciphered PIN and the file name of each data file 622-1, ..., on the basis of the data file creating data stored in the floppy disk 76, then transfers the generated dedicated file access command to the dedicated file access unit 636 of the IC card 6 via the data communication mechanisms 710 and 61 upon issuing the IC card 6, after the dedicated file creation instructing unit 723 transferred the dedicated file creating command to the IC card 6.

The personal computer 72 of the card issuing apparatus 7 according to this embodiment is provided the enciphering unit 725 which enciphers the PINs of the data files 622-1, ... supplied from the floppy disk 76 in order to write she enciphered PINs of the data files 622-1 into the dedicated file 624, and the decoding unit 726 which deciphers the enciphered PINs read out from the dedicated file 624 when PIN verification is carried out, as described later.

Upon checking a PIN of the IC card 6, the dedicated file access instructing unit 724 according to this embodiment functions to generate a dedi-

cated file access command including the master PIN in order to read out data from the dedicated file 624 of the IC card 6 which holds therein the PIN to be verified, then transfer the generated dedicated file access command to both the master PIN matching unit 635 and the dedicated file access unit 636 via the data communication mechanisms 710 and 61.

When the data (that is, data of a pair of the enciphered PIN and the file name of the data file 622-1) is read out from the dedicated file 624 of the IC card 6 in response to the dedicated file access command from the dedicated file access instructing unit 724 upon verification of the PIN of the IC card 6, the data file access instructing unit 722 according to this embodiment generates a data file access command including the PIN read out (that is, the PIN deciphered by the decoding unit 726), then transfers the generated data file access command to both the PIN matching unit 632 and the data file access unit 633 via the data communication mechanisms 710 and 61 so as to give an instruction to the IC card 6 to verify the correctness of the data file 622-1 corresponding to the PIN read out.

The printer 73 has a PIN printing mechanism 78. When the data file access instructing unit 724 verifies the correctness of the PIN read out by the dedicated file access instructing unit 724 upon PIN verification, the PIN printing mechanism of the printer 73 printouts the PIN that is a result of the verification on a predetermined printing paper 75 in response to an instruction from the personal computer 7.

To issue the IC card 6 according to this embodiment, the following process is performed, with the IC card 6 being inserted in the IC card reader/writer of the card issuing apparatus 7.

The dedicated file creating instructing unit 723 converts the dedicated file creating data (including the master PIN) stored in the floppy disk 77 into data in a format for a program incorporated in the IC card 6 to generate a dedicated file creating command including the master PIN, then transfers the generated dedicated file creating command to the dedicated file creating unit 634 of the IC card 6 via the data communication mechanisms 710 and 61.

In the IC card 6, when receiving the dedicated file creating command, the dedicated file creating unit 634 sets a control information unit 623-0 including a master PIN, a point and a file name of the dedicated file 624 in the directory area 623 in the storage unit 62 in response to the dedicated file creating command so as to create the dedicated file 624 in the file area 621 in the storage unit 62.

The data file creation instructing unit 721 in the card issuing apparatus 7 converts the data file creating data (including PINs for a card owner) stored the floppy disk 67 into data in a format for a program incorporated in the IC card 6 to generate a data file creating command including the PINs, then transfers the generated data file creating command to the data file creating unit 631 of the IC card 6 via the data communication mechanisms 710 and 61.

In the IC card 6, when receiving the data file creating command, the data file creating unit 631 sets a control information units 623-1 each including a PIN, a point and a file name of the data file 622-1 in the directory area 623 in the storage unit 62 in response to the data file creating command to create the data files 622-1 in the file area 621 in the storage unit 62.

Next, PINs are extracted from the data file creating data stored in the floppy disk in the card issuing apparatus 7, then enciphered to be an enciphered PINs. The dedicated file access instructing unit 724 then converts the data of a pair of the enciphered PIN and the file name of each data file 622-1 into data in a format for a program incorporated in the IC card 6 to generate a dedicated file access command including the above data, then transfers the generated dedicated file access command to the dedicated file access unit 636 of the IC card 6 via the data communication mechanisms 710 and 61.

When receiving the the dedicated access command, the dedicated file access unit 636 of the IC card 6 writes the enciphered PINs of the respective data files 622-1 in the dedicated file 624 such that the enciphered PIN of the data file 622-1 corresponds to its file name in response to the dedicated file access command, as shown in FIG. 6.

Through the above process, the IC card 6 is issued, going into a state to be able to receive general application services. At that time, the PINs for each card owner are managed in the dedicated file 624 in the IC card 6 that is possessed by the card owner.

To receive a general application service through the IC card, the owner inserts the IC card 6 into the terminal apparatus or the host computer providing the predetermined application. The terminal apparatus or host computer gives a data file access command to the IC card 6 to cause the IC card 6 to perform an access process (write/read) on each data file 622-1, ... in the storage unit 62.

More specifically, when receiving the data file access command from the terminal apparatus, host computer or the like, the PIN matching unit 632 makes a judgement as to whether a PIN of the data file 622-1 to be accessed (held in the control information unit 623-1 in the directory area 623) is in agreement with a PIN included in the data file access command supplied from the outside (i.e., a

PIN inputted by the card owner through the terminal apparatus, host computer or the like).

When the PIN matching unit 632 makes a judgement that the above two PINs are in agreement, the data file access unit 633 performs an access process (write/read) on the objective data file 622-1. When the PIN matching unit 632 successively draws the same conclusion predetermined times that the above two PINs are in disagreement, the PIN matching unit 632 locks the objective data file 622-1 so that the data file 622-1 is prohibited from being used.

Meanwhile, if the card owner forgets the PIN of his or her own IC card 6 after the issue of the IC card 6, it is possible to verify the PIN by carrying out the following process with the IC card 6 being inserted in card reader/writer 71 of the card issuing apparatus 6, according to this embodiment.

Namely, in order to read out data in the dedicated file 624 in the IC card 6, the dedicated file access instructing unit 724 generates a dedicated file access command including the master PIN, then transfers the generated dedicated file accessing command to both the master PIN matching unit 635 and the dedicated file access unit 636 of the IC card 6 via the data communication mechanisms 710 and 61.

On the side of the IC card 6, the master PIN matching unit 635 compares the master PIN (held in the control information unit 623-0 in the directory area 623) of the dedicated file 624 with the master PIN included in the dedicated file access command when receiving the dedicated file access command.

If the mater PIN matching unit 635 makes a judgement that the two master PINs are in agreement, the dedicated file accessing unit 636 reads out the data of a pair of the enciphered PIN and the file name of a data file from the dedicated file 624, then transfers it to the dedicated file access instructing unit 724 of the card issuing apparatus 7 via the data communication mechanisms 61 and 710.

The dedicated file access instructing unit 724 next makes the decoding unit 726 decipher the enciphered PIN, then reports data of a pair of the deciphered PIN and the file name of the data file to the data file access instructing unit 722.

When receiving the report form the dedicated file access instructing unit 724, the data file accessing instructing unit 722 generates a data file access command including the deciphered PIN, then transfers the generated data file accessing command to both the PIN matching unit 632 and the data file access unit 633 of the IC card 6 via the data communicating mechanisms verification on the correctness of the data file 622-1 corresponding to the PIN read out.

On the side of the IC card 6, the PIN matching unit 632 checks as to whether the PIN (held in the control information unit 623-1 in the directory area 623) of the data file 622-1 to be verified is in agreement with the PIN included in the data file accessing command supplied from the card issuing apparatus 7, in response to the data file accessing command to verify the correctness of the PIN.

If the PIN matching unit 632 draws a conclusion that the two PINs are in agreement, it is judged that the correctness of the data file 622-1 corresponding to the PIN read out has been verified. At that time, if the data file 622-1 that is an object of the correctness verification is in a locked state, the PIN matching unit the IC card 6 into a normal state.

When the correctness of the IC card 6 is verified in the above manner, the PIN printing mechanism 78 of the printer 73 printouts the deciphered PIN read out by the dedicated file access instructing unit 724 on the predetermined printing paper 75, then the verification of the PIN finishes.

According to the first embodiment, the data consisting of a pair of the PIN and file name of each data file 622-1, ... is set in the file area 621 in the storage unit 62 of the IC card 6. The PINs in each IC card 6 are managed by and within the IC card itself, without need for management of the PINs by the host computer, thereby largely reducing a burden for PIN management on the entire IC card system.

The data in the dedicated file 624 cannot be read out without the master PIN known by only the system manager. Even if the master PIN leaked out except the system manager, the PINs of the data files 622-1, ... do not leak out as they are, since each of the PIN of the data file 622-1 is enciphered. So long as the manner to encipher the PINs of the data files 622-1 is kept in secret, each PIN cannot be solved.

It is therefore possible to securely prevent the PINs from being out and avoid a problem in security, even if the PINs are managed by and within the IC card 6.

Further, upon verification of the PINs in the case of an accident, the dedicated file access instructing unit 724 of the card issuing apparatus 7 generates a dedicated file accessing command including the master PIN, then transfers the command to the IC card 6. The manner of this verification is quite simple and can reduce inconvenience to the card user.

According to this embodiment, when the PIN is verified, the data file access instructing unit 722 transfers a data file accessing command to the IC card 6 to verify the correctness of the PIN read out from the dedicated file 624, and the PIN matching unit 632 executes a PIN matching to make sure the

correctness of the data file 622-1 corresponding to the PIN read out, thereby increasing the reliability in the PIN verification.

In the above embodiment, the data file creating data and the dedicated file creating data are supplied from the floppy disks 76 and 77, respectively. It is also possible to input the data file creating data and the dedicated file creating data through a keyboard or the like.

(c) Description of Second Embodiment

Referring now to FIG. 8, an IC card 8 according to a second embodiment comprises a storage unit 81 and a control unit 82.

The storage unit 81 has a tree structure, including a file area 811 holding data to be processed by various application programs (i.e., programs incorporated in a terminal apparatus, a host computer, etc.) by files therein, and a directory area 813 holding control information about each of data files 812 in the file area 811 therein.

In the directory area 813, a file name, a personal identification number (PIN), a position (i.e., and the like of each data file 812 held in the file area 811 are written.

The IC card 8 according to this embodiment is additionally provided with a recovery information unit 815 in the data file 812 in the file area 811 in the storage unit 81, into which the control unit 82 writes recovery information obtained every time the data file 812 is updated, as shown in FIG. 9.

More concretely, as shown in FIG. 10, record numbers #1 to #n are allocated to an application area 814 to store data to be processed by one of various application programs (i.e., data of the balance in the case of a cashless card), and record numbers after #n + 1 and later are allocated to the recovery information unit 815.

As shown in FIG. 10, a start serial number (the number of processed items) obtained when the data file 812 is opened is written in the record number #n + 1 in the recovery information unit 815, an end serial number (the number of processed items) obtained when the data file 812 is closed is written in the record number #n + 2, and restoration data consisting of a record number whose data has been updated (i.e., an updated record number in the application area 814) and unupdated data at the same record number (i.e., data before subjected to the updating in the application area 814) obtained when the data file 812 is updated is written in the record numbers after #n + 3 and later. According to this embodiment, check serial numbers (the number of items) are respectively attached, as recovery information, before and after the start serial number, the restoration data and the end serial number held in the recovery information unit 815.

As shown in FIG. 9, there are set, in the directory area 813 in the storage unit 81 as control information of each data file 812, information as to whether the recovery information unit 815 is additionally provided in a data file 812 (that is, information about the presence of the recovery information in a data file 812), information about a relative position of the recovery information unit 815 in the data file 812, if the recovery information 815 exists in the data file 812 (that is, the leading record number of the recovery information unit 815), and a size and a number of the recovery information unit 815.

The control unit 82 according to this invention is adapted to manage the data files 812 in the file area 811 in the storage unit 81 on the basis of the control information held in the directory area 813 in the storage unit 81, comprising a data file opening unit 821, a data file updating unit 822, a data file closing unit 823, a start serial number obtaining unit 824, a restoration data obtaining unit 825, an end serial number obtaining unit 826, a restoration data initializing unit 827, a system failure detecting unit 828, a start serial number restoring unit 829, an end serial number restoring unit 830 and a data restoring unit 831.

When receiving an OPEN command (an open instruction) from an application program 9 incorporated in a terminal apparatus or the like into which the IC card 8 is inserted, the data file opening unit 821 opens a data file 812 designated by the open command, on the basis of the control information in the directory area 813 in the storage unit 81.

When receiving a WRITE command (an updating instruction) from the application program 9 after the data file 812 has been opened, the data updating unit 822 updates data in the data file 812 having been opened by the data file opening unit 821.

When receiving a CLOSE command (a closing instruction) from the application program 9 after the data file 812 has been opened, the data file closing unit 823 closes the data file 812 having been opened by the data file opening unit 821.

The start serial number obtaining unit 824 obtains a start serial number (whose initial value is 0) by adding 1 to the latest start serial number when the data file 812 is opened by the data file opening unit 821. The start serial number obtaining unit 824 also attaches check serial numbers (whose initial value is 1, incremented by 1 every time the data file 812 is opened) before and after the start serial number, and writes both the start serial number and its check serial numbers as recovery information in the record number #n + 1 of the recovery information unit 815.

The restoration data obtaining unit 825 acquires restoration data including a record number to be updated and unupdated data at the same record number when the data file 812 is updated by the data file updating unit 822. The restoration data obtaining unit 825 also attaches check serial numbers before and after the restoration data, and writes the restoration data and its check serial numbers as recovery information in the record number on and after #n + 3, in order.

If the data file updating unit 822 executes plural times of updating process on the same record number between an open and close of the data file 812, the restoration data obtaining unit 825 does not write data into the recovery information unit 815 after the second updating process and later.

The end serial number obtaining unit 826 obtains an end serial number (whose initial value is 0) by adding 1 to the latest start serial number when data file closing unit 823 closes the data file 812. The end serial number obtaining unit 826 also attaches check serial numbers (whose initial values are 1, incremented by 1 every time the data file 812 is closed) before and after the end serial number, and writes both the end serial number and its check serial numbers as recovery information into the record number #n + 2 in the recovery information unit 815.

Before the restoration data having been acquired by the restoration data obtaining unit 825 has been written into the recovery information unit 815, the restoration data initializing unit 827 initializes the restoration data that is being stored in the recovery information unit 815 (more specifically, setting the restoration data all at 0 and the check serial numbers all at 1).

When receiving an OPEN command (an open instruction) from the application program 9, the system failure detecting unit (a failure detecting means) 828, firstly, looks up the control information relating to a data file 812 designated by the OPEN command held in the directory area 813 in the storage unit 81. If the data file 812 has a recovery information unit 815 therein, the system failure detecting unit 828 detects a failure having occurred in the last process on the basis of the start serial number, the end serial number and the check serial numbers attached before and after these start serial number and end serial number, respectively, written as recovery information in the recovery information unit, through following procedure shown in FIG. 12.

When the check serial numbers attached before and after the start serial number are in disagreement, the system failure detecting unit 828 makes a judgement that a failure occurred when the data file opening unit 821 opened the data file in the last process, outputting a demand to reprocess the last process to the application program 9 and a demand to restore the start serial number to the start serial number restoring unit 829.

When receiving a demand to restore the start serial number form the system failure detecting unit 828, the start serial number restoring unit 829 restores the start serial number stored in the recovery information unit 815 to the one at the time of two updating processes earlier.

When the check serial numbers attached before and after the start serial number are in agreement but the check serial numbers attached before and after the end serial number are in disagreement, the system failure detecting unit 828 makes a judgement that a failure occurred when the data file closing unit 823 closed the data file in the last process, and outputs a demand to restore the end serial number to the end serial number restoring unit 830.

When receiving the demand to restore the end serial number from the system failure detecting unit 828, the end serial number restoring unit 830 restores the end serial number being stored in the recovery information unit 815 to the one at the time of the last process.

When the check serial numbers attached before and after the start serial number and the check serial numbers attached before and after the end serial number are in agreement but the start serial number and the end serial number are in disagreement, the system failure detecting unit 828 judges that a failure occurred when the data file updating unit 822 updated the data file in the last process, and outputs a demand to perform once again the last process to the application program 9 and a demand to restore the data in the data file 812 to the data restoring unit 831.

When receiving the demand to restore the data in the data file 812 from the system failure detecting unit 828, the data restoring unit 831 restores the data in the application area 814 in the data file 812 on the basis of the restoration data stored in the recovery information unit 815.

The data restoring unit 831, according to this embodiment, comprises a restoration data effectiveness detecting unit 832 and a restoration data writing unit 833.

The restoration data effectiveness detecting unit 832 judges that the restoration data is effective when the check serial numbers attached before and after the restoration data are in agreement. The restoration data writing unit 833 writes the unupdated data of the restoration data having been judged to be effective by the restoration data effectiveness detecting unit 832 as the data at the record number of the above restoration data in the application area 814 of the data file 812.

According to this embodiment, the recovery information is acquired in the IC card 8 in a normal updating process instructed by the application program 9 through following procedure shown in FIG. 11.

When receiving an OPEN command from the application program 9, the system failure detecting unit 828 of the IC card 8 detects a failure that has occurred in the last process (Step S1).

If a system failure is not detected, or a system failure is detected and a recovery process (a data recovery process) is executed by the restoring unit 829-831, the restoration data initializing unit 827 initializes the restoration data that has been written in the recovery information unit 815 in the last process by setting all the restoration data at 0 and all the check serial numbers at 1 (Step S2). (Meanwhile, a manner for the system failure detecting and the data restoring will be described later, referring to FIG. 12.).

The start serial number obtaining unit 824 obtains a check serial number and a start serial number by adding 1 to the latest start serial number (whose initial value is 0) and the check serial numbers (whose initial value is 1), and writes the new start serial number and its check serial numbers attached before and after the start serial number as recovery information in the record number #n + 1 in the recovery information unit 815 (Step S3).

Thereafter, the data file opening unit 821 opens the data file 812 designated by the open command on the basis of the control information in the directory area 813 in the storage unit 81 (Step S4).

After the data file 812 is opened, the restoration data obtaining unit 825 obtains restoration data including a record number to be updated and unupdated data at this record number every time the application program 9 supplies a WRITE command (an updating instruction), and writes the restoration data and its check serial numbers attached before and after the restoration data in the record number on and after #n + 3 in the recovery information unit 815, in order (Step S5). In the case where the data file updating unit 822 executes the updating process plural times on the same record number between an open and close of the data file 812, the restoration data will not be written into the recovery information unit 815 after the second updating process and later.

After the restoration data obtaining unit 825 has obtained the restoration data, the data file updating unit 822 updates the data in the data file 812 (Step S6).

While the application program 9 supplies a WRITE command (an updating instruction), the process in the steps S5 and S6 is repeated.

When receiving a close command from the application program 9, the end serial number obtaining unit 826 obtains an end serial number and its check serial numbers by adding 1 to the latest end serial number (whose initial value is 0) and the check serial numbers (whose initial value is 1) as same as in the step S3. The end serial number with the check serial numbers attached before and after the end serial number are written into the record number #n + 2 in the recovery information unit 815 as recovery information (Step S7).

Thereafter, the data file closing unit 823 closes the data file 812 (Step S8).

In the above manner, the recovery information is written in the recovery information additionally provided in the data file 812 in the file area 811 in the storage unit 81 every time the control unit causes the data file 812 to be updated.

A manner to detect a system failure in the last process by the system failure detecting unit 828 and a restoring process (a data recovery process) when a system failure is detected in the IC card 8 carried out in Step 1 shown in FIG. 11 will be now described in more detail, referring to FIG. 12.

When receiving an OPEN command from the application program 9, a reference is made to the control information about a designated data file 812 in the directory area 813. If the data file 812 has a recovery information unit 815 therein, the system failure detecting unit 828 detects a system failure having occurred in the last process, on the basis of the recovery information (i.e., a start serial number, an end serial number and check serial numbers attached before and after the start serial number and the end serial number) stored in the recovery information unit 815.

Namely, a judgement is first made as to whether the check serial numbers attached before and after the start serial number are in agreement or not (Step S11). If the check serial numbers are in disagreement, it is judged that a system failure occurred when the data file was opened by the data file opening unit 821 in the last process, then a demand to reprocess the last process and a demand to restore the start serial number are outputted to the application program 9 and the start serial number restoring unit 829, respectively.

The start serial number restoring unit 829 restores the start serial number stored in the recovery information unit 815 to the one at the time of two updating processes earlier (Step S12). Then in the step S2 shown in FIG. 11, the application program 9 once more carries out the process that should have been done in the last process in response to the reprocessing demand from the system failure detecting unit 828.

On this occasion, since a system failure occurred when the data file was opened in the last

process, it can be assumed that no updating process has been performed on the data file 812, and thus the restore data and end serial number in the recovery information unit 815. Therefore, only the start serial number is restored to the one at the time of two updating processes earlier without carrying out the data restoring (data recovery), and a demand for the reprocessing is given to the application program 9. As this, the program of the last process can be appropriately executed in the recovery information unit 815.

If it is judged in the Step S11 that the check serial numbers attached before and after the start serial number are in agreement, the next judgement is made as to whether the check serial numbers attached before and after the end serial number are in agreement or not (Step S13). If the check serial number attached before and after the end serial number are in disagreement, it is judged that a failure occurred when the data file was opened by the data file opening unit 823 in the last process, and a demand to restore the end serial number is outputted to the end serial number restoring unit 830.

The end serial number restoring unit 830 then restores the end serial number stored in the recovery information unit 815 to the one at the time of the last process (Step S14). Then the procedure advances to the step S2 in FIG. 11, the application program 9 carried out the process of this time.

On this occasion, since a failure occurred when the data file was closed in the last process, it can be assumed that the data file 812, and the start serial number and the restoration data in the recovery information unit 815 have been properly updated in the last process. Therefore, it is possible to execute the process of this time with a proper recovery information unit 25 only if the end serial number is restored to the one at the time of the last process.

If it is judged in the step S13 that the check serial numbers attached before and after the end serial number is in agreement, the next judgement is made as to whether the start serial number and the end serial number are in agreement or not (Step S15). If in disagreement, it is judged that a failure has occurred during the data file updating operation carried out by the data file updating unit 822 in the last process, then a demand to reprocess the last process and a demand to restore the data in the data file 812 are outputted to the application program 9 and the data restoring unit 831, respectively.

When receiving a command to restore the data in the data file 812 from the system failure detecting unit 828, the data restoring unit 831 restores the data in the application area 814 in the data file 812 on the basis of the restoration data stored in the recovery information unit 815 (Step S16). The procedure then advances to the step S2 in FIG. 11, where the application program 9 carries out once more the operation that should have been done in the last process.

On this occasion, since a failure has occurred during the updating process in the last process, the application area 814 in the data file 812 is restored to the state at the time of two updating processes earlier (the state before the failure has occurred), then a reprocess is demanded to the application program 9, thereby reexecuting the last process on the data file 22 that is in a state at the time of two updating processes earlier.

When the data restoring unit 831 restores the data, the restoration data writing unit 833 writes only the restoration data whose check serial numbers attached before and after the same have been judged by the restoration data effectiveness detecting unit 832 to be in agreement into the data file 812.

It is therefore possible to restore the data without using the restoration data (whose check serial numbers are in disagreement) that a system failure has occurred in the course of writing the same in the last process.

In the case where the start serial number is in agreement with the end serial number in the step S15, it is judged to be normal (Step S17), the procedure advances to the step S2 in FIG. 11, where the process of this time is executed by the application program 9.

Next, operation of the IC card according to this embodiment will be described referring to FIGS. 13 through 16, where a content of practical data in the recovery information unit 815 is shown to explain the operation.

The recovery information unit 815 immediately after the issue of the IC card 8 is in a state where the start serial number, the end serial number and the restoration data are all set at 0, and the check serial numbers attached before and after the start serial number, the end serial number and the restoration data are all set at 1, as shown in FIG. 13A.

Assuming that a WRITE instruction for, for example, the record numbers #10, #8 and #11 of the data file 812 from the application program 9 is successively executed between an open and close of the data file 812 without a break due to a system failure, in the first updating process done on the IC card 8.

In which case, the start serial number obtaining unit 821 and the end serial number obtaining unit 823 obtain "1" and "1" as a start serial number and an end serial number, respectively, to write them into the recovery information unit 815. At the same time, the restoration data obtaining unit 825 also obtains, for example, "#10,3030", "#08,F1F1"

and "#11,1010" as restoration data, and writes them into the recovery information unit 815. Before and after each data, check serial numbers "2" are attached, as shown in FIG. 13B.

Here, "3030", "F1F1" and "1010" in the restoration data are unupdated data at the record numbers #10, #08 and #11 in the application area 814.

Thereafter, the restoration data initializing unit 827 initializes to make all the restoration data to be "0" and their check serial numbers to be "1" when the second updating process is performed on the IC card 8. Assuming that after the initialization has been executed by the restoration data initializing unit 827, a WRITE instruction is successively executed two times on the record number #02 of the data file 812, and this updating process is executed between an open and close of the data file 812, without a break due to a system failure.

In which case, the start serial number obtaining unit 821 and the end serial number obtaining unit 823 obtain "2" and "2" as a start serial number and an end serial number, respectively, and write them into the recovery information unit 815. At the same time, the restoration data obtaining unit 825 obtains, for example "#02,4040" as restoration data, and also writes it into the recovery information unit. Check serial numbers "3" are attached before and after the start serial number, the end serial number and the restoration data, and also written into the recovery information unit 815.

In the case where updating process is executed a plurality of times on the same record number in the between an open and close of the data file 812, the restoration data obtaining unit 825 does not write the restoration data into the recovery information unit 815 after the second process and later. "4040" written as unupdated data of the restoration data is the first unupdated data in relation to the record number #02" in the application area 814.

On the third updating process done on the IC card 8, the restoration data initializing unit 827 first executes initialization. The application program 9 next gives a WRITE instruction for, for example, the record number #10 and #08 in the data file 812. Now assuming that a system failure occurred after restoration data in connection with the record number #08 was obtained, as shown in FIG. 14.

In which case, the start serial number obtaining unit 821 obtains a start serial number "3" and its check serial numbers "4", and writes them into the recovery information unit 815. The end serial number obtaining unit 823, however, cannot obtain an end serial number and its check serial numbers since a system failure has occurred before receiving a CLOSE instruction. As a result, the end serial number "2" and the check serial number "3" at the time of the last process remain in the recovery information unit 815.

The restoration data obtaining unit 825 obtains, for example, "#10,3030" and "#08,F1F1" as restoration data, and successively writes them with their check serial numbers "4" into the recovery information unit. The updating process is terminated.

If the application program 9 executes an updating process on the IC card 8 under the above condition, the system failure detecting unit 828 operates in response to an OPEN instruction from the application program 9, and makes a judgement that a system failure has occurred during the last updating process since the check serial number attached before and after the start serial number and the check serial numbers attached before and after the end serial number are individually in agreement but the start serial number and the end serial number are in disagreement (referring to Step S15 in FIG. 12). The system failure detecting unit 828 thus outputs a demand to perform a reprocess to the application program 9 and a demand to restore the data in the data file 812 to the data restoring unit 831 (referring to the Step 16 in FIG. 12).

When the data restoring unit 831 restores the data, the restoration data effectiveness detecting unit 832 verifies the effectiveness of restoration data from whether check serial numbers attached before and after the restoration data are in agreement or not. In the example shown in FIG. 14, the check serial numbers attached before and after two restoration data are all "4", being thus in agreement. The two restoration data are thus judged to be effective. The restoration data writing unit 833 writes the restoration data "3030" and "F1F1" into the respective record numbers #10 and #08 in the application area 814 in the data file 812 on the basis of the above two restoration data whose effectiveness has been verified.

On the contrary, assuming that upon performing the third updating process on the IC card 8, a WRITE instruction from the application program 9 is executed on the record numbers, for example, #03 and #02 in the data file 812, and a system failure occurs while the restoration data in connection to the record number #2 is being obtained, as shown n FIG. 15.

In which case, the start serial number obtaining unit 821 obtains a start serial number "3" and its check serial numbers "4", and write them into the recovery information unit 815. The end serial number obtaining unit 823, however, cannot obtain an end serial number and its check serial numbers since a system failure has occurred before receiving a CLOSE instruction. As a result, the end serial number "2" and the check serial number "3" at the

time of the last process remain in the recovery information unit 815.

The restoration data obtaining unit 825 obtains "#03,3232" as restoration data and its check serial numbers "4". However, since a system failure has occurred in the course of obtaining "#02,2222" as the second restoration data, check serial numbers "4" are attached before the restoration data "#02,2222" while check serial numbers "3" attached after the restoration data in the last process still remains even at the end of the process.

Under such condition, if an updating process is executed on the IC card 8 from the application program 9, the system failure detecting unit 828 operates in response to an OPEN instruction from the application program 9, and makes a judgement that a system failure has occurred during the last updating process since a check serial number attached before and after the start serial number and a check serial number attached before and after the end serial number are individually in agreement, but the start serial number and the end serial number are in disagreement (referring to the step 15 in FIG. 12). The system failure detecting unit 828 thus outputs a demand to once more perform the last process to the application program 9 besides a demand to restore the data in the data file 812 to the data restoring unit 831, as same as in the example shown in FIG. 14 (referring to the Step S16 in FIG. 12).

Here, when the data restoring unit 831 restores the data, the restoration data effectiveness detecting unit 832 verifies the effectiveness of the restoration data by making a judgement as to whether the check serial numbers attached before and after the restoration data are in agreement or not. In the example shown in FIG. 15, the check serial numbers attached before and after the first restoration data are both "4", being thus in agreement. But, the check serial numbers attached before and after the second restoration data are "4" and "3", being thus in disagreement.

For this, the first restoration data is judged to be effective, but it is judged that a system failure has occurred in the course of obtaining the second restoration data so that the updating process on the record number #02 in the application area 814 has not been completed. Therefore, the restoration data writing unit 833 conducts the writing on the basis only the first restoration data.

Through the above process, it is possible to restore the application area 814 to a state before a system failure has occured from a state of the system failure as shown in FIG. 16B, by writing the restoration data "3232" to the record number #3 in teh application area 814 of the data file 812, respectively.

As above, according to the second embodiment of this invention, a start serial number and an end serial number in the recovery information unit 815 are compared with each other. If the start serial number and the end serial number are in disagreement, it is possible to detect that a system failure has occurred in the course from an open to close of the data file 812. Check serial numbers attached before and after each data in the recovery information unit 815 are also compared with each other. If the check serial numbers are in disagreement, it is possible to detect that a system failure occurred in the course of writing the start serial number, the restoration data or the end serial number into the recovery information unit 815, further detect effectiveness of each data stored in the recovery information unit 815 and conflicting data generated due to the system failure without using a BCC, with certainty.

Before the restoration data has been obtained by the restoration data obtaining unit 825 is written into the recovery information unit 815, the restoration data of the last process stored in the recovery information unit 815 is initialized by the restoration data initializing unit 827. This initializing operation can prevent the restoration data of the last process from remaining in the recovery information unit when the new restoration data is written in the recovery information unit, and also can surely avoid to erroneously detect a system failure.

In the case where the data file updating unit 822 executes updating process more than once on the same record number between an open and close of the data file 812, the restoration data obtaining unit 825 does not write the restoration data in the recovery information unit 815 after the second updating and later, whereby it is possible to always hold the data before the data file is opened (i.e., the data before the updating) as the restoration data in the recovery information unit 815 at the same record number. Therefore, the state within the IC card 8 after the system failure has occurred can be effectively restored to the state before the system failure has occurred, on the basis of the data in the recovery information unit 815.

Since information about the presence of the recovery information unit 815 and information about a relative position of the recovery information unit 815 in a data file if the recovery information unit 815 exists in the data file are both set in the directory area 813 in the storage unit 81, it is possible to make a judgement as to whether predetermined data should be written into the recovery information unit 815 or data should be restored on the basis of the data stored in the recovery information unit, only by referring to the directory area 813 in the storage unit 81 from the control unit 82.

If the data file 812 has the recovery information unit therein, the system failure detecting unit 828 detects a system failure that has occurred in the last process, on the basis of the start serial number, the end serial number and the check serial numbers attached before and after the start serial number and the end serial number in the recovery information unit 815, in response to an OPEN instruction form the application program 9. This enables an automatic detection of conflicting data occurring due to a system failure within the IC card 8 without using a BCC.

According to a result of detection carried out by the system failure detecting unit 828, the start serial number restoring unit 829, the end serial number restoring unit 830 and the data restoring unit 831 can automatically repair and restore the recovery information unit 815 or the application area 814, whereby a configuration of the system can be simplified and inconvenience to the owner upon restoring the data can be mitigated to a considerable degree.

When the data restoring unit 813 repairs the data, the restoration data writing unit 833 writes only the effective restoration data whose check serial numbers attached before and after the same have been judged to be in agreement by the restoration data effectiveness judging unit 832 into the application area 814 in the data file 812. In consequence, it is possible to restore, certainly and effectively, the data in the application area 814 to the state before a system failure has occurred, without using restoration data that the system failure has occurred in the course of the writing (i.e., data whose check serial numbers are in disagreement).

In the IC card 8 according to the second embodiment, a terminal used to send and receive information to and from the outside (a contact and a data communication mechanism) is omitted in the drawings.

In the second embodiment state above, there is no need to add the recovery information unit 815 to all data file 812. As shown in FIG. 9, it is possible to omit the recovery information unit 815 in the data file 812 that needs no data recovery.

In the second embodiment, description has been made by way of an IC card as a card type storage medium. This invention is, however, not limited to the above examples, but adaptable to another type of card type storage medium, for example, an optical card, bringing the same effect as the above examples.

It is also possible to form a card type storage medium having a function of the IC card 6 according to the first embodiment, along with a function of the IC card 8 according to the second embodiment. In this case, the advantages of the above two embodiments can be realized in one card type storage medium.

Claims

1. A card type storage medium comprising a storage unit having a file area (21) holding data in each file as a unit and a directory area (23) holding therein control information units (231) each including a PIN of a data file in said file area and a control unit (3) managing data files (22) in said file area (21) in said storage unit (2) on the basis of said control information units (231) in said directory area (23) in said storage unit (2), said control unit (3) allowing an access process on a data file (22) by said control unit (3) only when a PIN held in said control information unit (231) in said directory area (23) in said storage unit (2) is in agreement with a PIN inputted from outside, the storage medium further comprising:

   a dedicated file (24) being set in said file area (21) in said storage unit (2) , said dedicated file (24) holding PINs of the data files (22) held in said respective control information units (231) in said directory area (23) in said storage unit (2) and file names of the data file (22) such that the PIN and the file name of each data file (22) correspond to each other;

   another control information unit (232) being set in said directory area (23) in said storage unit (2), said control information unit (232) holding a master PIN of said dedicated file (24).

2. A card type storage medium according to claim 1, wherein the PINs of the respective data files (22) are enciphered to be held in said dedicated file (24).

3. A card type storage medium comprising:

   a storage unit (2) having a file area (21) holding data in each file as a unit and a directory area (23) holding therein control information units (231) each including a PIN of each data file (22) in said file area (21);

   a control unit (3) managing data files (22) in said file area (21) in said storage unit (2) on the basis of said control information units (231) in said directory area (23) in said storage unit (2) , said control unit (3) comprising;

   a data file creating means (31) , in response to a data file creating command from outside, setting a control information unit (231) for a data file (22) including a PIN of said data file (22) to create said data file (22) in said file area (21) in said storage unit (2) according to said data file creating command;

a PIN matching means (32), in response to a data file access command to gain an access to the data file (22) created by said data file creating means (31) from the outside, making a judgement as to whether the PIN of said data file (22) to be accessed according to said data file access command held in said control information unit (231) in said directory area (23) in said storage unit (2) is in agreement with a PIN included in said data file access command supplied from the outside;

a data file accessing means (32) executing an access process on the data file (22) to be accessed when said PIN matching means (32) judges that said two PINs are in agreement;

a dedicated file creating means (34), in response to a dedicated file creating command from the outside, setting a control information unit (232) for said dedicated file (24) including a master PIN for said dedicated file (24) to create said dedicated file (24) in said file area (21) in said storage unit (2) according to said dedicated file creating command;

a master PIN matching means (35), in response to a dedicated file access command to gain an access to said dedicated file (24) created by said dedicated file creating means from the outside, making a judgement as to whether the master PIN of said dedicated file (24) held in said control information unit (232) in said directory area (23) in said storage unit (2) is in agreement with a master PIN included in said dedicated file access command supplied from the outside; and

a dedicated file access means (36) executing an access process on said dedicated file (24) when said master PIN matching means (35) makes a judgement that the above two master PINs are in agreement;

upon issuing said IC card, said dedicated file accessing means (36) writing the PINs of the data files (22) held in said respective control information units (231) in said directory area (23) in said storage unit (2) into said dedicated file (24) such that the PIN and file name of each data file (22) correspond to each other according to a dedicated file accessing command supplied from outside after said dedicated file creating means (34) created said dedicated file.

4. A card type storage medium according to claim 3, wherein PINs for the respective data files (22) are enciphered and held in said dedicated file (24).

5. A card type storage medium issuing apparatus issuing a card type storage medium (10), said card type storage medium (10) comprising a storage unit (2) having a file area (21) holding data in each file as a unit and a directory area (23) holding therein control information units (231) each including a PIN of a data file (22) in said file area (21) and a control unit (3) managing data files (22) in said file area (21) in said storage unit (2), comprising:

a data file creation instructing means (41) setting a control information unit (231) for a data file (22) including a PIN of the data file (22) in said directory area (23) in said storage unit (2), generating a data file creating command including the PIN of the data file (22), and transferring it to said card type storage medium (10) in order to create the data file (22) in said file area (21) in said storage unit (2);

a data file access instructing means (42) generating a data file accessing command including a PIN of a data file (22) to be accessed, and transferring it to said card type storage medium (10) in order to gain an access to the data file (22) created in the file area (21) in said storage unit (2);

a dedicated file creation instructing means (43) setting a control information unit (232) for a dedicated file (24) including a master PIN of the dedicated file (24) in said directory area (23) in said storage unit (2), generating a dedicated file creating command including the master PIN, and transferring it to said card type storage medium (10) in order to create the dedicated file (24) in said file area (21) in said storage unit (2);

a dedicated file access instructing means (44) generating a dedicated file accessing command including the master PIN of the dedicated file (24), and transferring it to said card type storage medium (10) in order to gain an access to the dedicated file (24) created in said file area (21) in said storage unit (2);

upon issuing said card type storage medium (10), after said dedicated file creation instructing means (43) transferred a dedicated file creating command to said card type storage medium (10), said dedicated file access instructing means (44) generating a dedicated file accessing command including data of PINs and file names of the data files (22), and transfers it to said card type storage medium (10), in order to write the PINs and file names of the data files (22) held in said respective control information units (231) in said directory area (23) in said storage unit (2) such that the PIN and file name of each data file (22) correspond to each other.

6. A card type storage medium issuing apparatus according to claim 5, wherein when a PIN of said card type storage medium (10) is verified, said dedicated file access instructing means (44) generates a dedicated file access command including the master PIN, and transfers it to said card type storage medium (10) in order to read out data held in said dedicated file (24) in said file area (22) in said storage unit (2) of said card type storage medium (10) whose PIN is to be verified.

7. A card type storage medium issuing apparatus according to claim 6, wherein when data including a PIN and file name of a data file (22) held in said card type storage medium (10) is read out from said dedicated file (24) of said card type storage medium (10) in response to a dedicated file accessing command send out from said dedicated file access instructing means (44), said data file access instructing means (44) generates a data file accessing command including the PIN read out, and transfers it to said card type storage medium (10) so as to give an instruction to said card type storage medium (10) to verify correctness of the data file (22) corresponding to said PIN read out.

8. A card type storage medium issuing apparatus according to any one of claims 5 through 7, wherein said card type storage medium issuing apparatus further comprises an enciphering means enciphering the PINs of the data files (22) to be written in said dedicated file (24) in said card type storage medium (10) by said dedicated file access instructing means (44), and a deciphering means deciphering an enciphered PIN of a data file (22) read out from said dedicated file (24) in said card type storage medium (10) by said dedicated file access instructing means (44).

9. A card type storage medium comprising a storage unit (2) having a file area (21) holding data in each file as a unit and a directory area (23) holding therein control information about each data file (22) in said file area (21) and a control unit (5) managing data in said file area (21) in said storage unit (2) on the basis of said control information held in said directory area (23) in said storage unit (2) , said control unit (5) executing updating on a data file (22) in response to an instruction supplied from outside, the storage medium comprising:

a recovery information unit (25) provided in a data file (22) in said file area (21) in said storage unit (2), into which recovery information obtained every time said control unit (5) updates the data file (22) is written;

a start serial number obtained when the data file (22) is opened and an end serial number obtained when the data file is closed are written as recovery information into said recovery information unit (25).

10. A card type storage medium according to claim 9, wherein restoration data including a record number to be updated and unupdated data at said record number obtained when the data file (22) is updated are written as recovery information into said recovery information unit (25).

11. A card type storage medium according to claim 10, wherein check serial numbers are attached before and after the start serial number, the restoration data and the end serial number, respectively, as recovery information in said recovery information unit (25).

12. A card type storage medium according to any one of clams 9 through 11, wherein when updating is executed plural times on the same record number in the course from an open to close of a data file (22), restoration data is not written into said recovery information unit (25) on and after the second updating.

13. A card type storage medium according to any one of claims 9 through 11, wherein information about the presence of a recovery information unit (25) in a data file (22) and information about a relative position of said recovery information unit (25) in the data file (22) if said recovery information unit (25) exists in the data file (22) are set in said control information of the data file (22) in said directory area (23) in said storage unit (2).

14. A card type storage medium comprising a storage unit (2) having a file area (21) holding data in each file as a unit and a directory area (23) holding control information about each data file (22) in said file area (21) and a control unit (5) managing data in said file area (21) in said storage unit (2) on the basis of the control information in said directory area (23) in said storage unit (2) , said control unit (5) comprising a data file opening means (51) opening a data file (22) in said file area (23) in said storage unit (2) according to an opening instruction supplied from outside on the basis of the control information in said directory area (23) in said storage unit (2) in response to the opening instruction, a data file updating means

(52) updating data held in a data file (22) having been opened by said data file opening means (51) in response to an updating instruction supplied form the outside after the data file (22) has been opened, and a data file closing means (53) closing the data file (22) having been opened by said data file opening means (51) in response to a closing instruction supplied from the outside after the data file (22) has been opened, the storage medium comprising:

a recovery information unit (25) provided in a data file (22) in said file area (21) in said storage unit (2), said recovery information unit (25) holding therein recovery information obtained every time said control unit (5) updates the data file (22);

said control unit (5) further comprising;

a start serial number obtaining means (54) obtaining a start serial number when said data file opening means (51) opens a data file (22) to write the start serial number as recovery information into said recovery information unit (25); and

an end serial number obtaining means (56) obtaining an end serial number when said data file closing means (53) closes the data file (22) to write the end serial number as recovery information into said recovery information unit (25).

15. A card type storage medium according to claim 14, wherein said control unit (5) further comprises a restoration data obtaining means (55) obtaining restoration data including a record number to be updated and unupdated data at said record number when said data file updating means (52) updates said data file (22) to write it as recovery information into said recovery information unit (25).

16. A card type storage medium according to claim 15, wherein when said start serial number obtaining means (54), said restoration data obtaining means (55) and said end serial number obtaining means (56) obtain a start serial number, restoration data and an end serial number, respectively, check serial numbers are attached as recovery information before and after said start serial number, restoration data and end serial number, respectively, and are written into said recovery information unit (25).

17. A card type storage medium according to any one of clams 14 through 16, wherein said control unit (5) further comprises a restoration data initializing means initializing restoration data that has been written in said recovery information unit (25) in the last process, before new restoration data obtained by said restoration data obtaining means (55) is written into said recovery information unit (25).

18. A card type storage medium according to any one of claims 14 through 17, wherein if said data file updating means (52) executes updating plural times on the same record number in the course from an open to close of a data file (22), said restoration data obtaining means (55) avoids writing restoration data into said recovery information unit (25) on and after the second updating.

19. A card type storage medium according to any one of claims 14 through 18, wherein information about the presence of a recovery information unit (25) in a data file (22) and information about a relative position of said recovery information unit (25) in the data file (22) if said recovery information unit (25) exists in the data file (22) are set in the control information of each data file (22) in said directory area (23) in said storage unit (2).

20. A card type storage medium according to claim 14, wherein said control unit (5) further comprises a failure detecting means, in response to an open instruction to open a data file (22) supplied from outside, referring to control information about said data file (22) held in said directory area (23) in said storage unit (2), if said data file (22) has a recovery information unit (25), said failure detecting means detecting a failure that has occurred in the last process on the basis of a start serial number, an end serial number and check serial numbers attached before and after the start serial number and end serial number, respectively, written in said recovery information unit (25) as recovery information.

21. A card type storage medium according to claim 20, wherein said failure detecting means makes a judgement that a failure occurred when said data file opening means (51) opened a data file (22) in the last process if check serial numbers attached before and after a start serial number of said data file (22) are in disagreement.

22. A card type storage medium according to claim 21, wherein said failure detecting means outputs a demand to reprocess the last process and a demand to restore the start serial number if said failure detecting means detects

the a failure occurred when the data file (22) was closed in the last process.

23. A card type storage medium according to claim 22, wherein said control unit (5) further comprises a start serial number restoring means restores, in response to a restoring demand to restore a start serial number supplied form said failure detecting means, said start serial number written in said recovery information unit (25) into a start serial number at the time of two processes earlier.

24. A card type storage medium according to claim 20, wherein said failure detecting means makes a judgement that a failure occurred when said data file closing means (53) closed a data file (22) if check serial numbers attached before and after a start serial number of said data file (22) are in agreement but check serial numbers attached before and after an end serial number of said data file (22) are in disagreement.

25. A card type storage medium according to claim 24, wherein said failure detecting means outputs a demand to restore the end serial number of the data file (22) if said failure detecting means detects that a failure occurred when said data file (22) was closed in the last process.

26. A card type storage medium according to claim 25, wherein said control unit (5) further comprises an end serial number restoring means restores, in response to a demand to restore an end serial number of a data file (22) supplied from said failure detecting means, the end serial number written in said recovery information unit (25) to an end serial number at the time of the last process.

27. A card type storage medium according to claim 20, wherein said failure detecting means makes a judgement that a failure occurred when said data file updating means (52) updated a data file (22) if check serial numbers attached before and after a start serial number of the data file and check serial numbers attached before and after an end serial number of the same are individually in agreement but the start serial number and the end serial number are in disagreement.

28. A card type storage medium according to claim 22, wherein when said failure detecting means detects a failure that has occurred during the last updating process of a data file (22), said failure detecting means outputs a demand to reprocess the last process and a demand to restore data in said data file (22).

29. A card type storage medium according to claim 28, wherein said control unit (5) further comprises a data restoring means restores, in response to a demand to restore data in a data file (22) supplied from said failure detecting means, said data in said data file (22) on the basis of restoration data stored in said recovery information unit (25).

30. A card type storage medium according to claim 29, wherein said data restoring means comprises:

a restoration data effectiveness detecting means making a judgement that restoration data is effective if check serial numbers attached before and after said restoration data are in agreement; and

a restoration data writing means writing unupdated data of restoration data that has been judged to be effective by said restoration data effectiveness detecting means as data at a record number of said restoration data into a data file (22).

FIG. 1

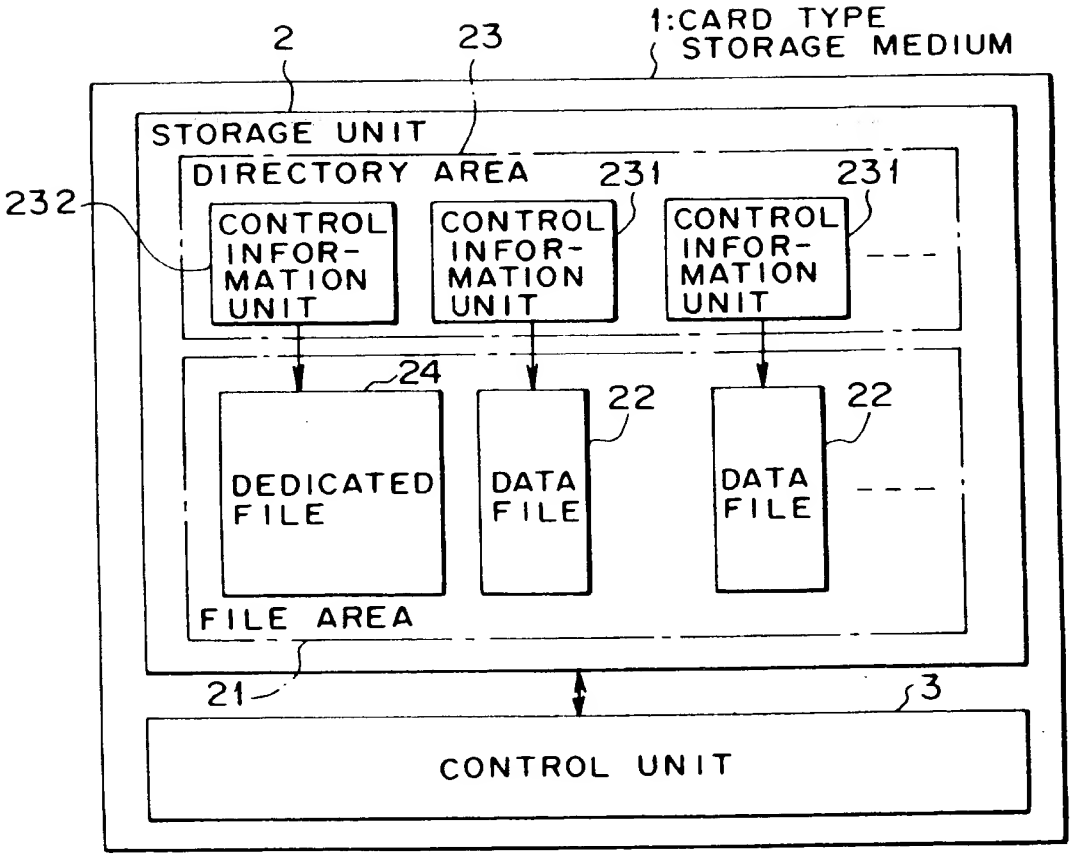


FIG. 2

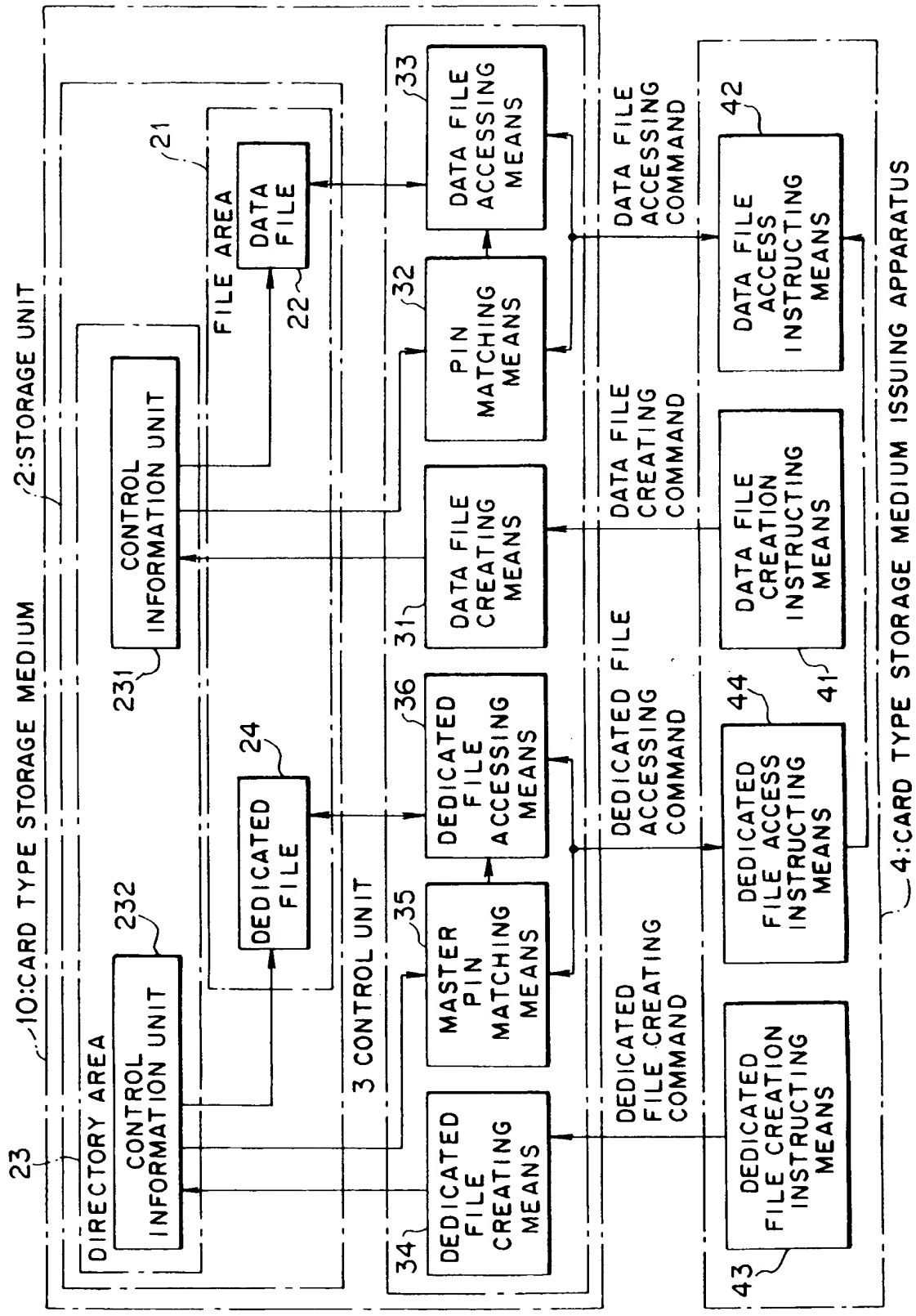


FIG. 3

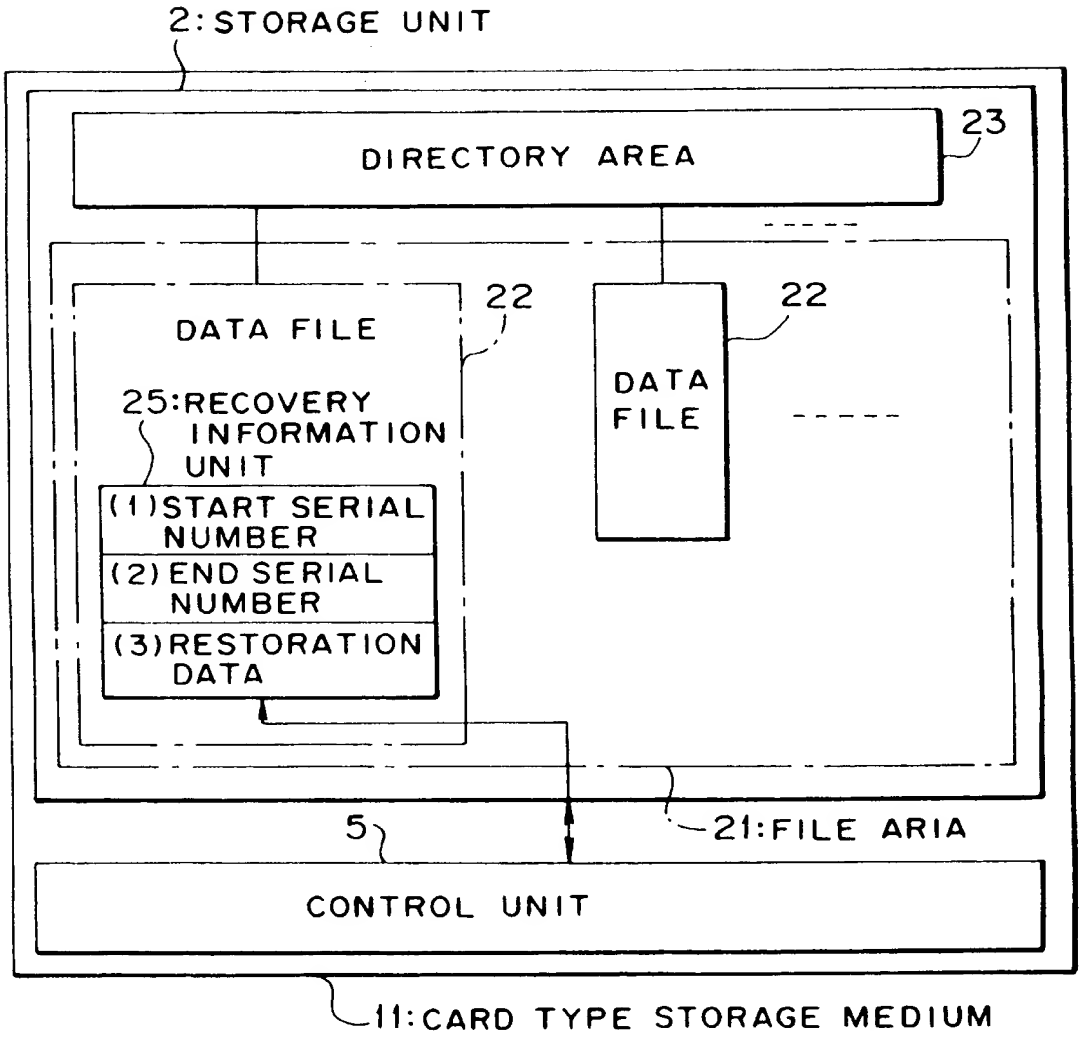


FIG. 4

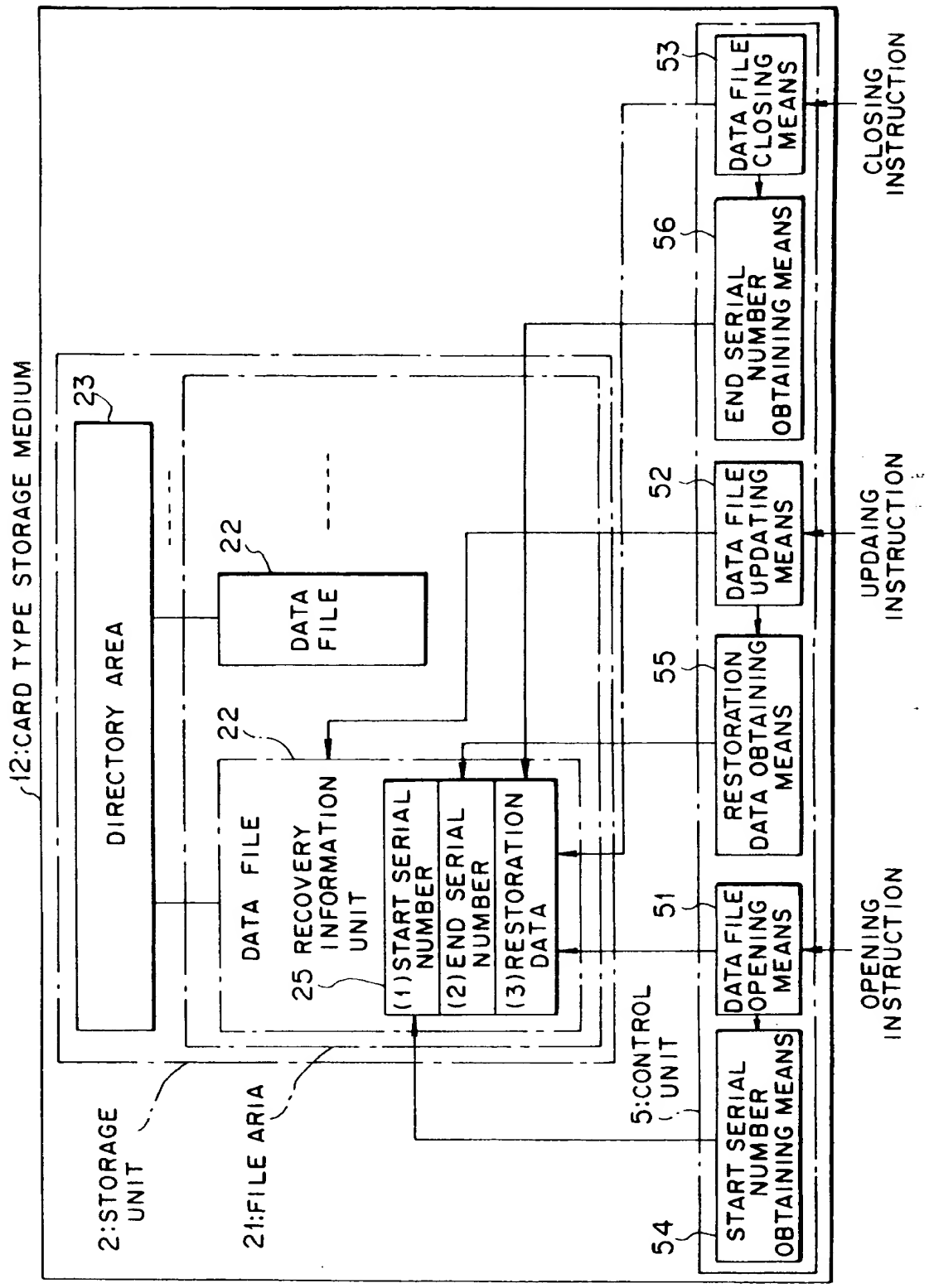


FIG.5

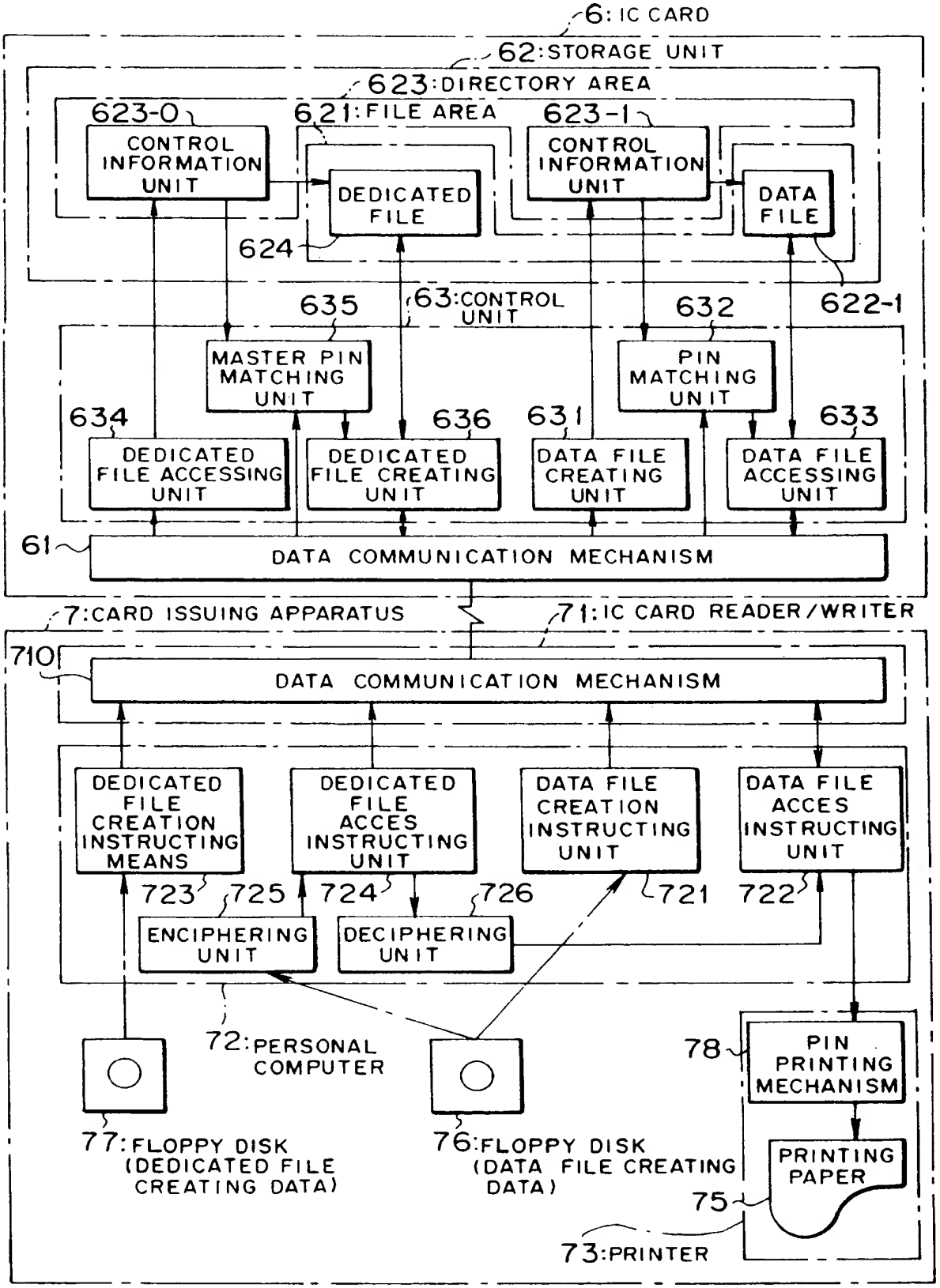


FIG.6

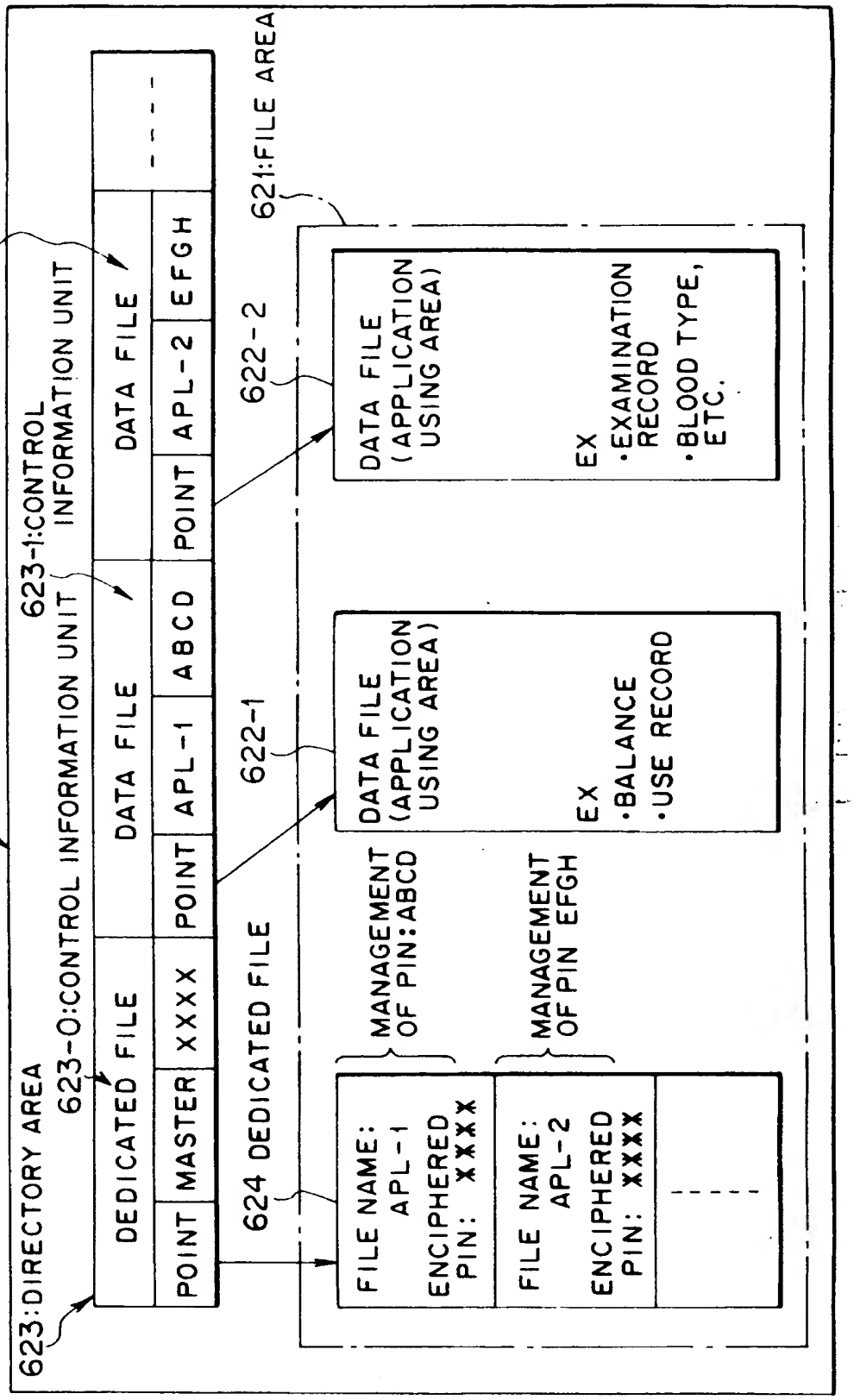


FIG. 7

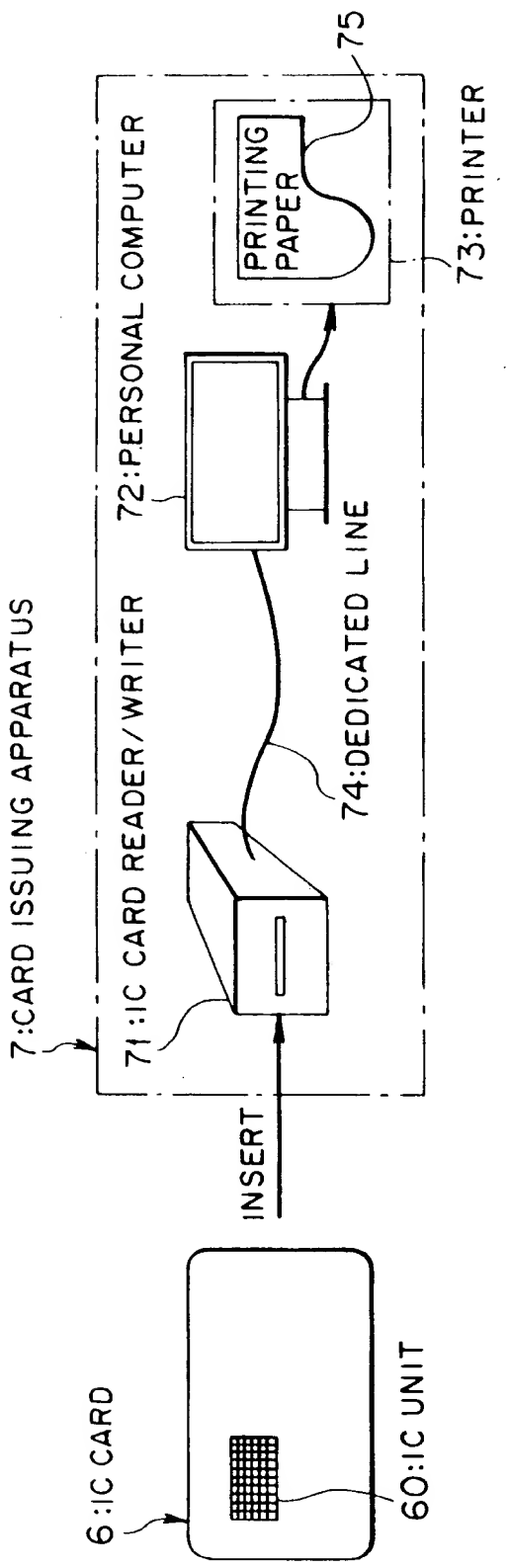


FIG.8

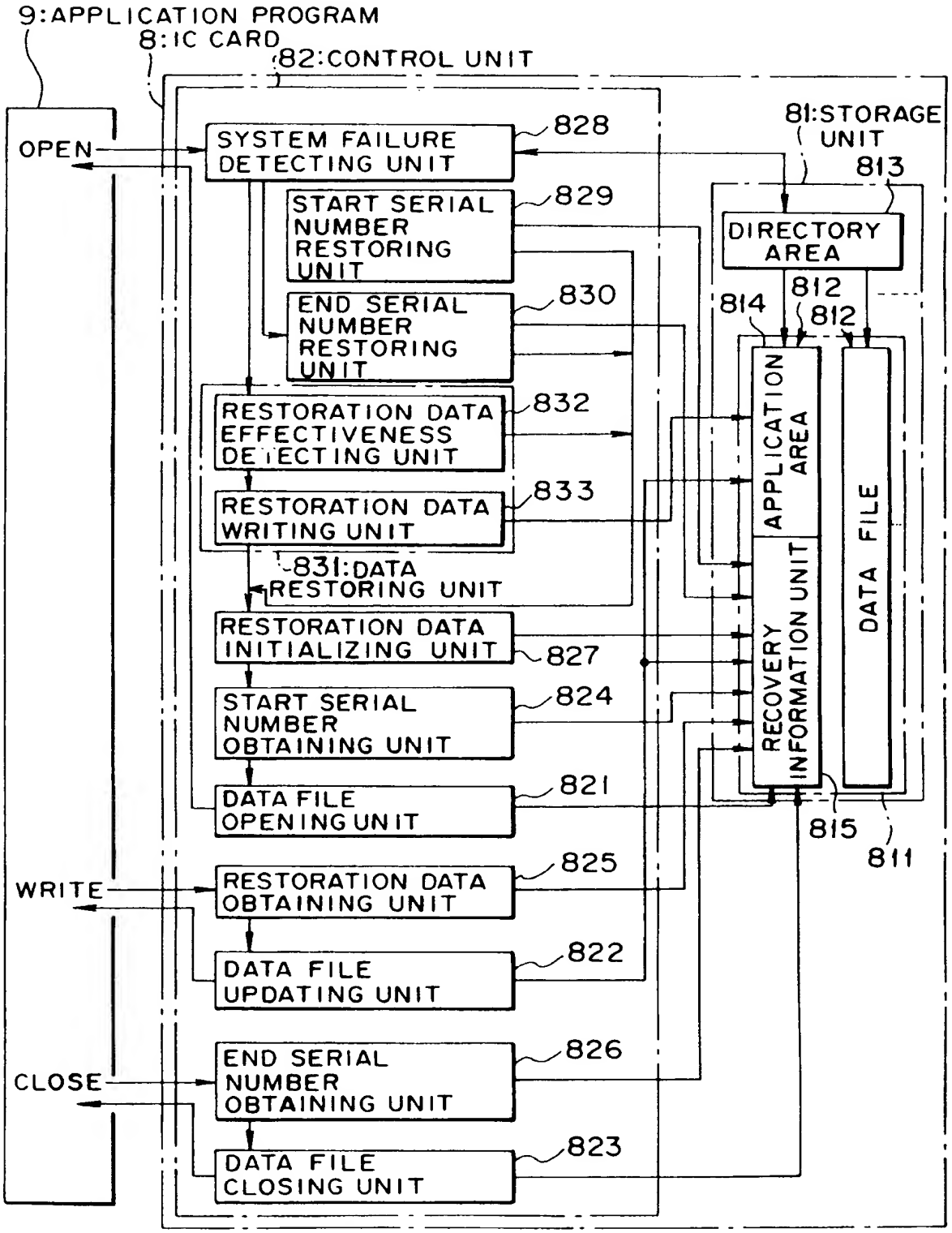


FIG. 9

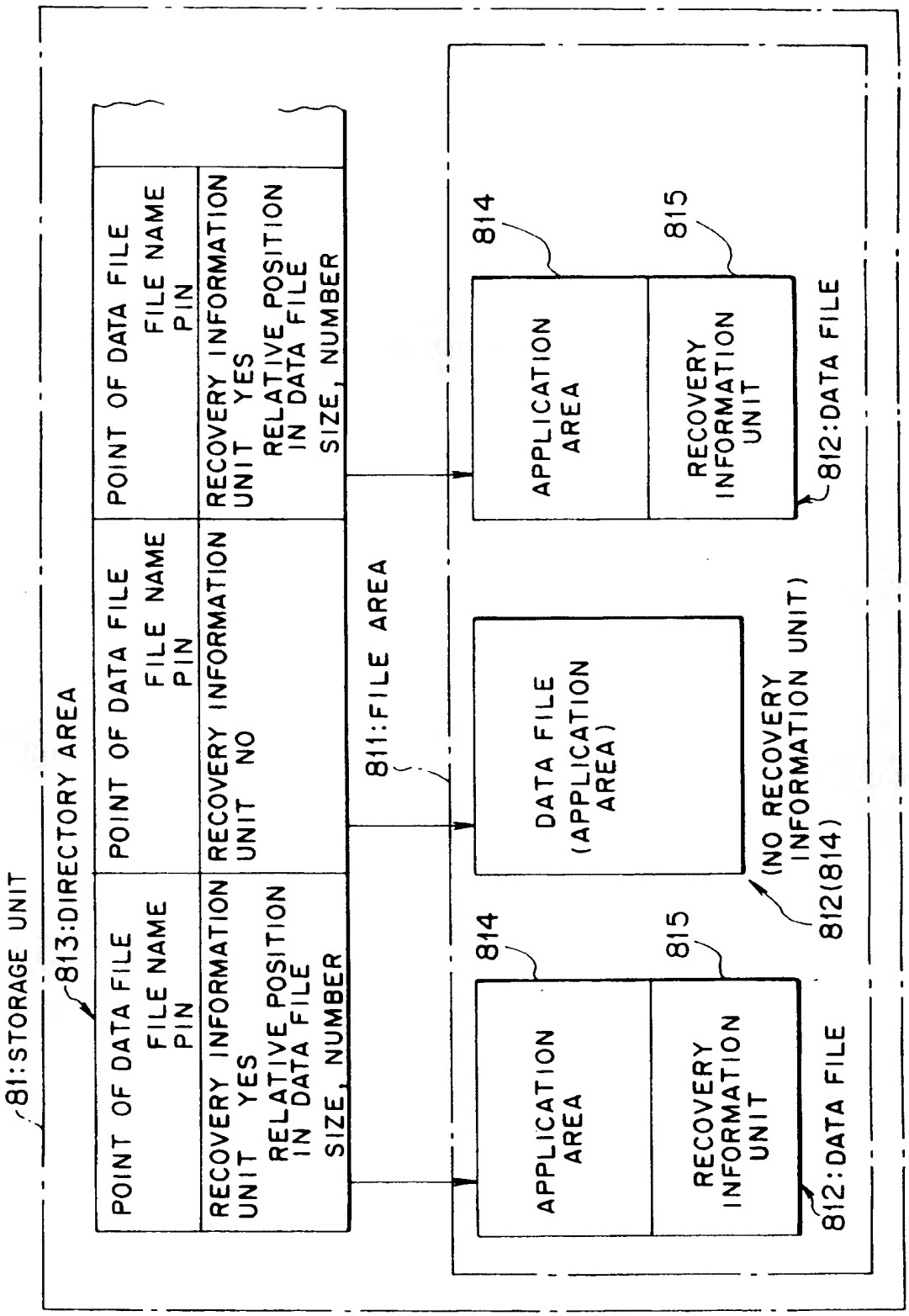


FIG. 10

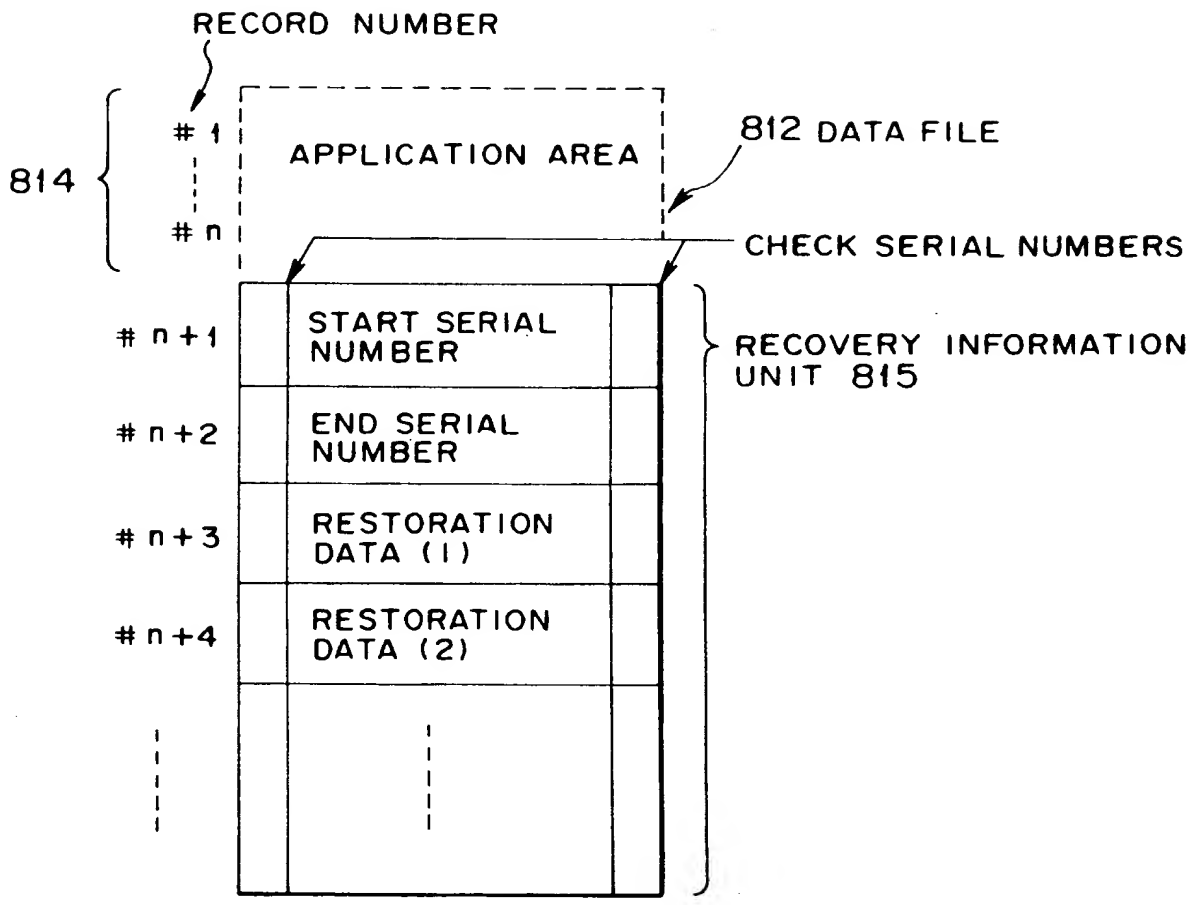


FIG. 11

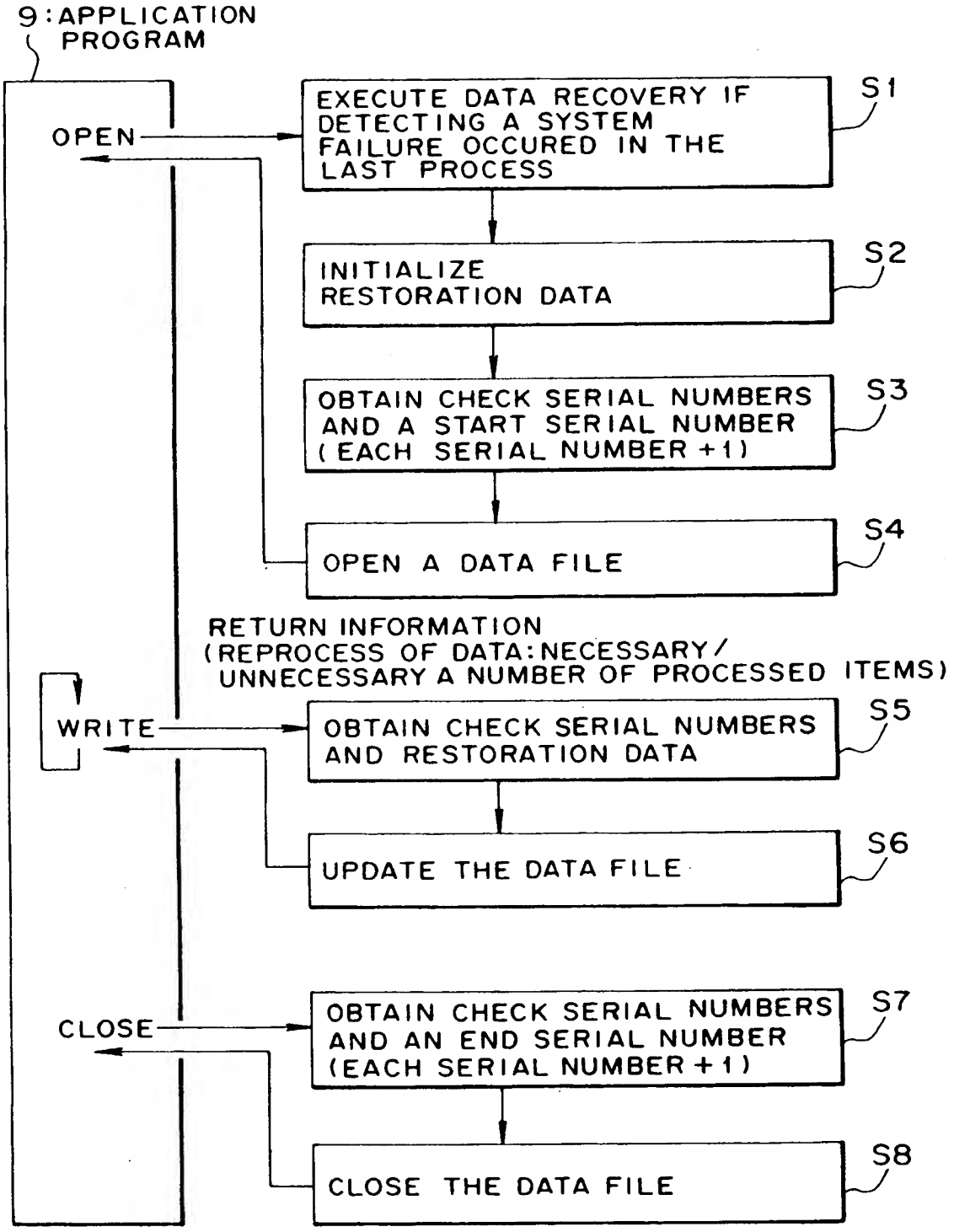


FIG. 12

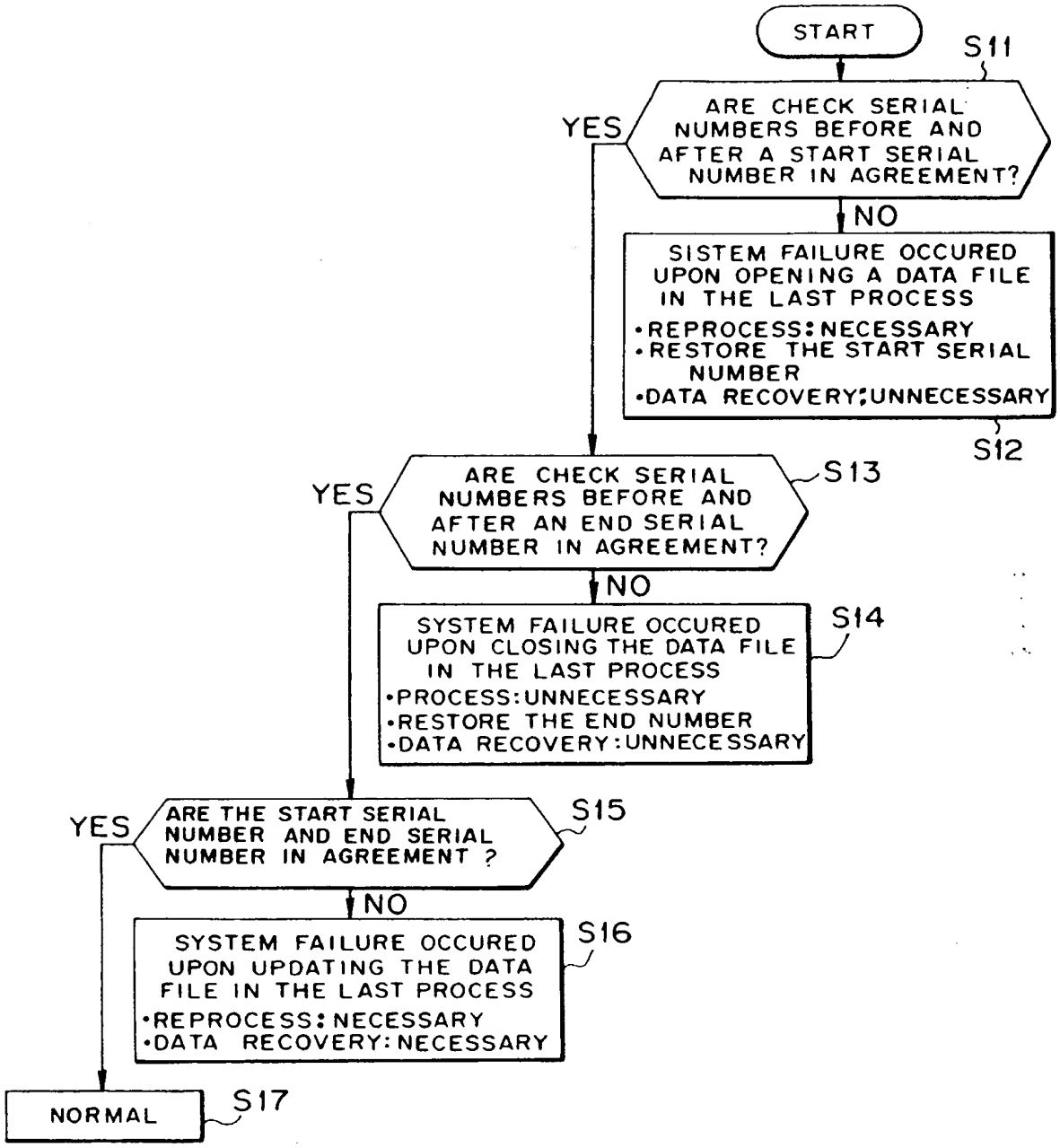


FIG. 13A

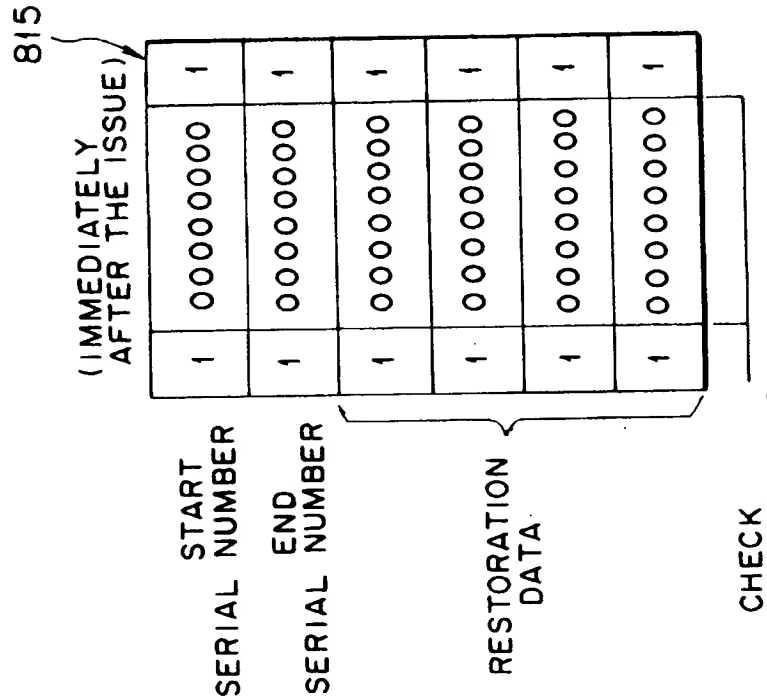


FIG. 13B

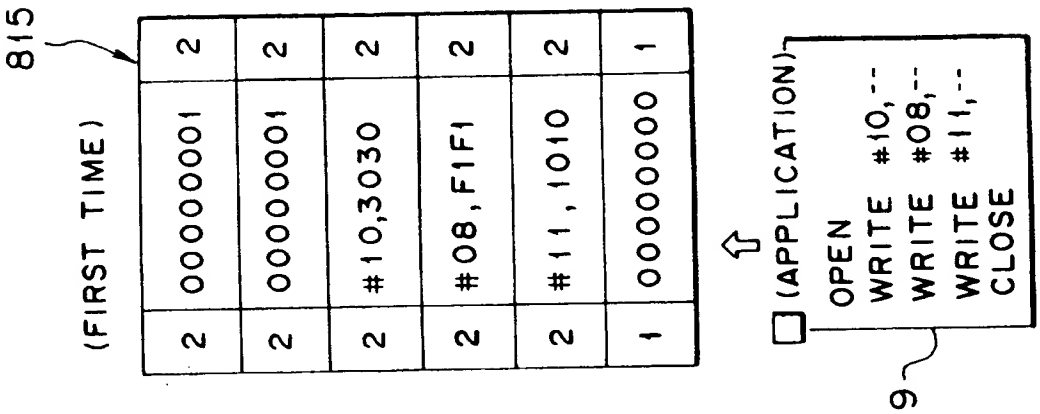


FIG. 13C

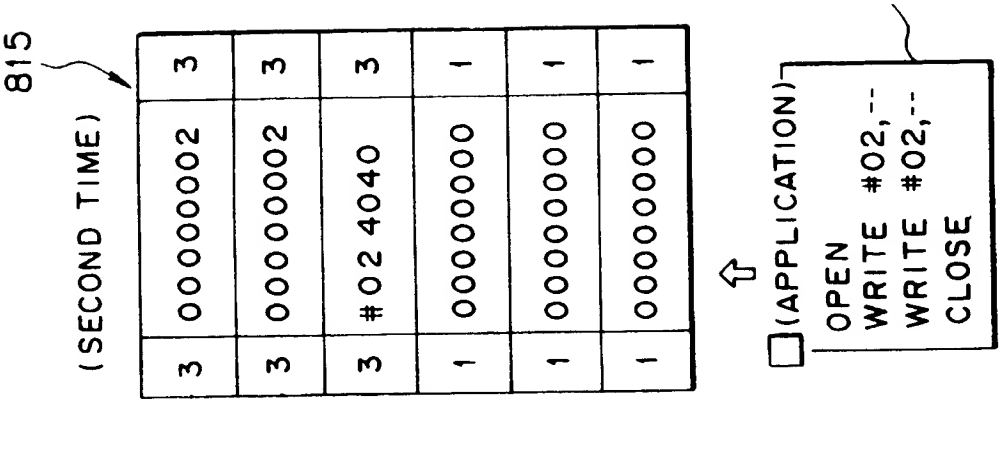


FIG.14

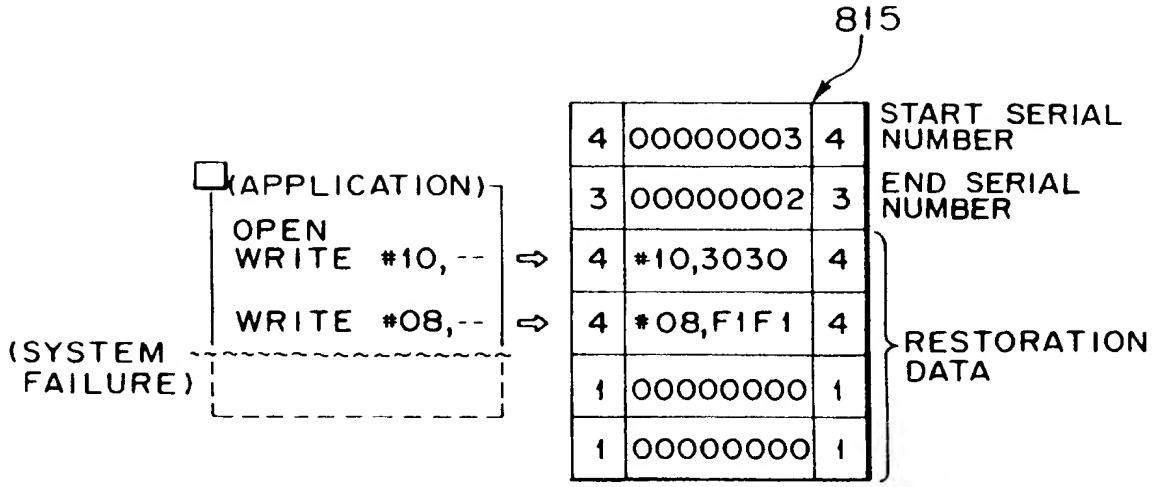


FIG.15

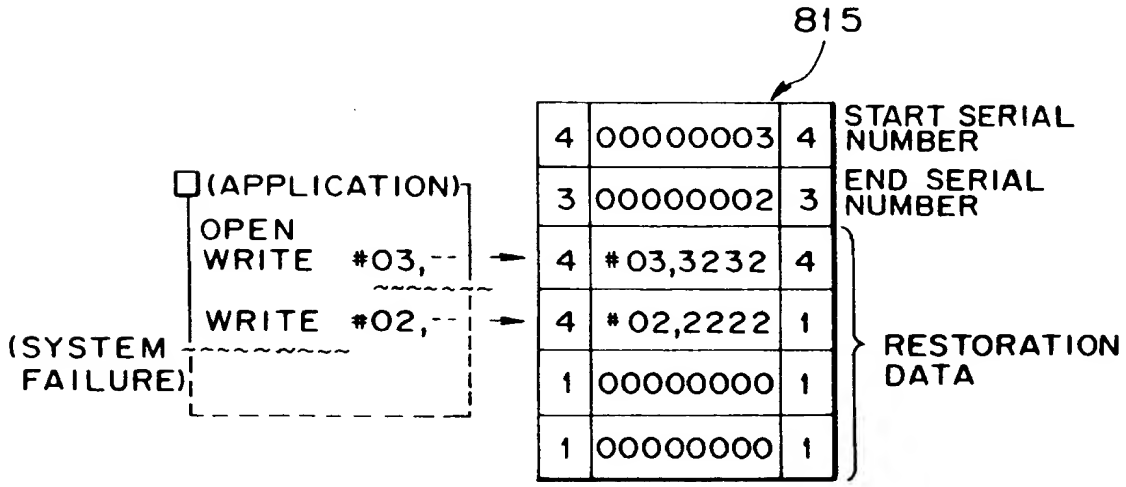


FIG.16A

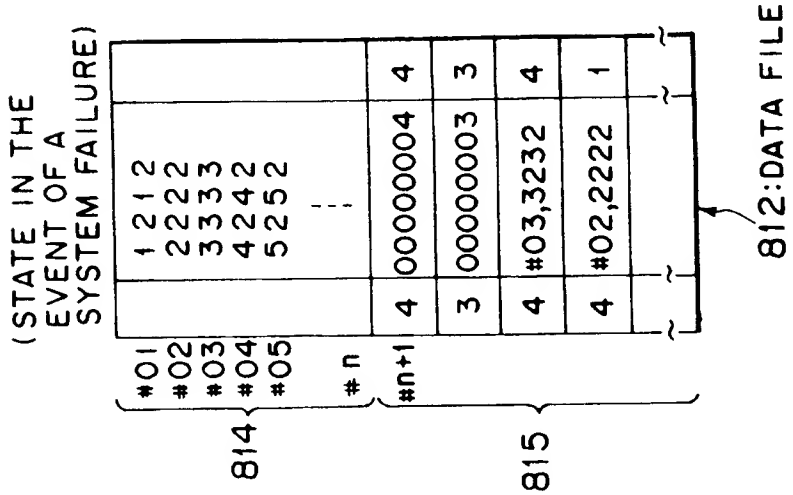


FIG.16B

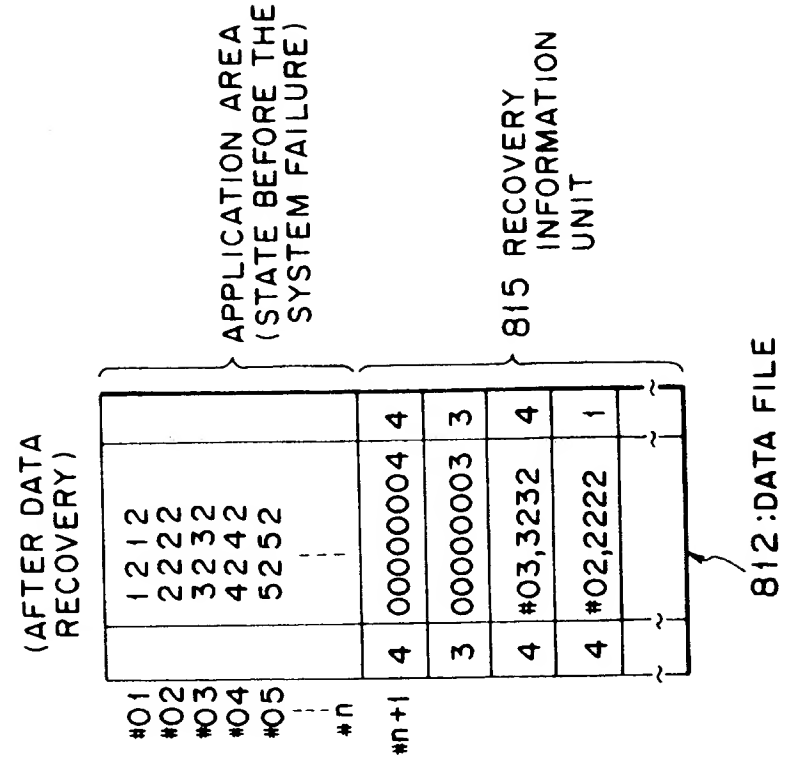


FIG. 17

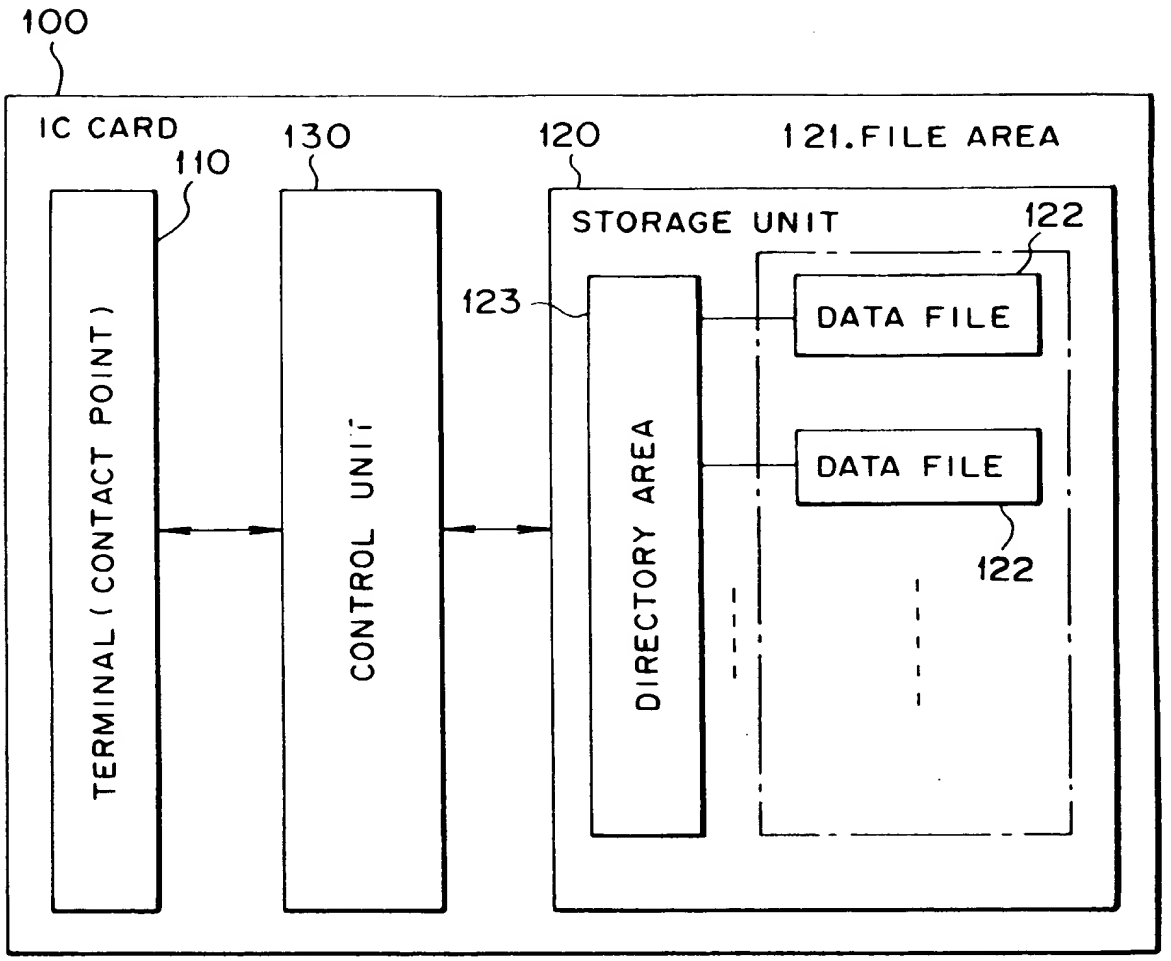


FIG. 18

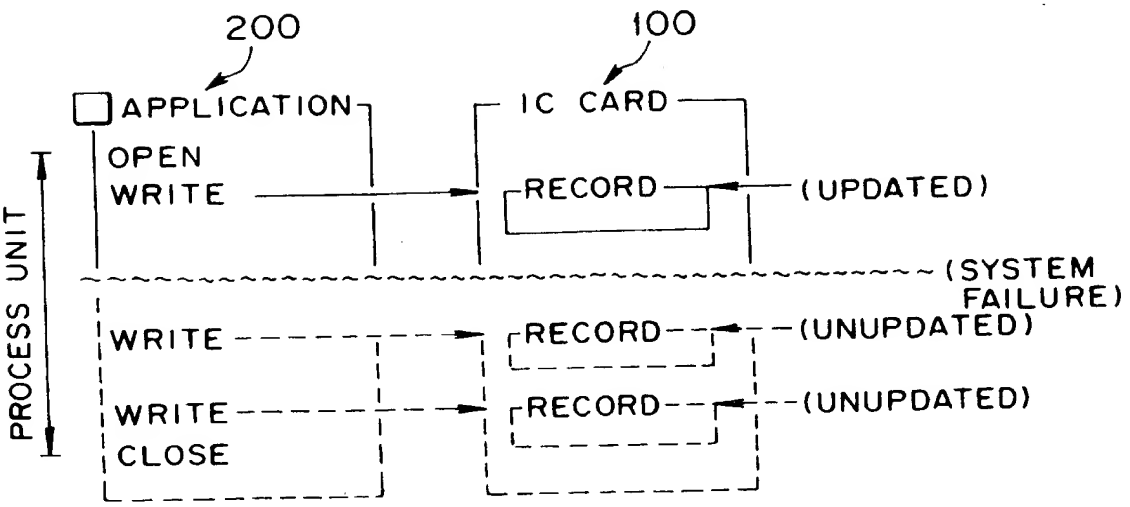
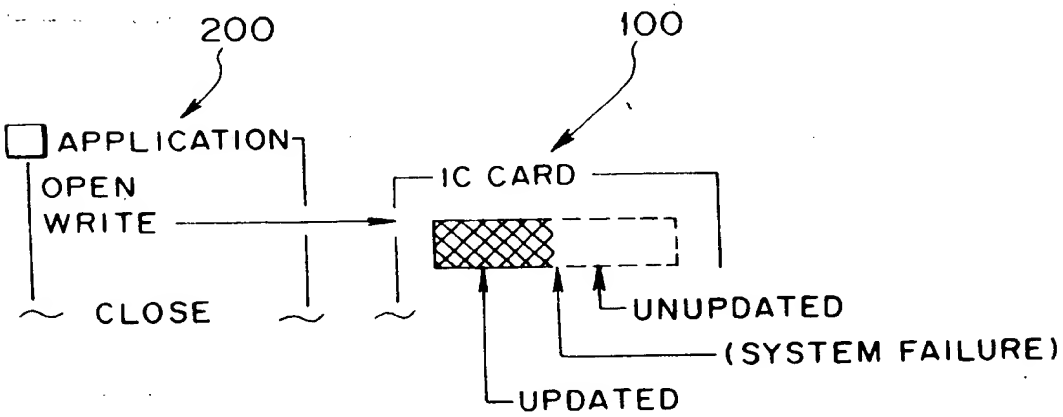


FIG. 19



[redacted] EUROPEAN PATENT OFF[redacted]

Patent Abstracts of Japan

PUBLICATION NUMBER : 09305734
PUBLICATION DATE : 28-11-97

APPLICATION DATE : 15-05-96
APPLICATION NUMBER : 08119906

APPLICANT : DAINIPPON PRINTING CO LTD;

INVENTOR : HARIMA HIROTSUGU;

INT.CL : G06K 19/07 G06F 9/06 G06K 17/00

TITLE : IC CARD, PROGRAM INTRODUCING METHOD AND PROGRAM EXECUTING METHOD

ABSTRACT : PROBLEM TO BE SOLVED: To provide an IC card, etc., in which plural application programs(AP) are stored and any arbitrary one of them can be selected/executed.

SOLUTION: Concerning a program introducing method for introducing the AP into the non-volatile memory of the IC card, address information is initially set to the prescribed address of the non-volatile memory and based on information provided in the AP specified by the address information, update processing for updating the address information is executed certain times equal with the number of AP. Then, the AP applied from the outside is written into the area, where the AP can be stored, specified by the updated address information and the information showing the number of AP is updated.